现代数学基础丛书 161

有限集上的映射与动态过程
——矩阵半张量积方法

程代展　齐洪胜　贺风华　著


科学出版社

(O-6191.01)

www.sciencep.com



ISBN 978-7-03-046376-0

科学数理分社
电 话：(010) 64011058
Email：lixin_kx@mail.sciencep.com

销售分类建议：高等数学

定 价：148.00 元

“十二五”国家重点图书出版规划项目

现代数学基础丛书 161

有限集上的映射与动态过程
——矩阵半张量积方法

程代展 齐洪胜 贺风华 著

科学出版社
北京

内容简介

应用矩阵半张量积这一新工具，本书研究有限集合之间的映射的表达和性质，以及有限集合上的动态系统的演化规律与控制. 内容分三部分：① 矩阵半张量积与有限集映射，包括矩阵半张量积；有限集映射的代数表示；命题逻辑与布尔函数、布尔多项式、布尔代数、布尔矩阵；逻辑函数的复合分解. ② 有限集上的动态系统，包括逻辑动态系统及其代数状态空间表示；逻辑控制系统的能控、能观性，干扰解耦，稳定性与镇定；逻辑系统辨识. ③ 有限博弈，包括非合作博弈；演化博弈；势博弈；有限博弈的空间分解；对称博弈；合作博弈；演化博弈的优化控制. 对于这些问题，本书利用新的工具，从新的角度审视，给出一系列新的结果.

本书所需要的预备知识仅为大学工科的微积分和线性代数，如果有一点常微分方程、线性系统理论及初等概率论的知识就更好了. 本书的阅读对象为离散数学、自动控制、计算机、系统生物学、博弈论及相关专业的高年级本科生、研究生、青年教师及科研人员.

图书在版编目（CIP）数据

有限集上的映射与动态过程：矩阵半张量积方法/程代展，齐洪胜，贺风华著. —北京：科学出版社，2015. 11

“十二五”国家重点图书出版规划项目

（现代数学基础丛书；161）

ISBN 978-7-03-046376-0

I. ①有…　II. ①程…　②齐…　③贺…　III. ①矩阵—乘法　IV. ①O151.21

中国版本图书馆 CIP 数据核字（2015）第 271160 号

责任编辑：李　欣　赵彦超／责任校对：张凤琴

责任印制：张　倩／封面设计：陈　敬

科学出版社 出版

北京东黄城根北街 16 号

邮政编码：100717

http://www.sciencep.com

三河市骏杰印刷有限公司 印刷

科学出版社发行　各地新华书店经销

*

2016 年 1 月第　一　版　开本：720 × 1000 1/16

2016 年 1 月第一次印刷　印张：25 1/2

字数：514 000

定价：148.00 元

（如有印装质量问题，我社负责调换）

作 者 简 介

程代展 中国科学院数学与系统科学研究院研究员. 1970 年毕业于清华大学, 1981 年于中国科学院研究生院获硕士学位, 1985 年于美国华盛顿大学获博士学位. 曾经担任过国际自动控制联合会 (IFAC) 理事 (Council Member), IEEE 控制系统学会(CSS) 执委 (Member of Board of Governors), *Int. J. Math Sys., Est. Contr*. (1991-1993), *Automatica* (1998-2002), *Asia J. Control*. (1999-2004)的 Associate Editor. *International Journal on Robust and Nonlinear Control* 的 Subject Editor, 国内杂志 *J. Control Theory and Application* 主编, 《控制与决策》的副主编及多家学术刊物的编委, 中国自动化协会控制理论专业委员会主任, IEEE CSS 北京分会主席等. 已经出版了 10 本专著, 发表了 230 多篇期刊论文和 120 多篇会议论文. 主要研究方向包括非线性控制系统、数值方法、复杂系统、布尔网络控制、基于博弈的控制等. 曾两次作为第一完成人获国家自然科学二等奖 (2008, 2014), 2011 年获 IFAC 所颁 *Automatica* (2008-2010) 最佳论文奖. 2006 年入选 IEEE Fellow, 2008 年入选 IFAC Fellow.

齐洪胜 中国科学院数学与系统科学研究院副研究员. 2008 年于中国科学院数学与系统科学研究院获博士学位. 主要研究方向包括非线性控制、复杂系统与控制、博弈论等. 2011 年获 IFAC 所颁 *Automatica* (2008-2010) 最佳论文奖, 2014 年获国家自然科学二等奖 (排名第二).

贺风华 哈尔滨工业大学航天学院教授、博士生导师. 2004 年于哈尔滨工业大学获博士学位. 主要研究方向包括飞行器制导与控制、博弈与控制等. 曾获国防科技进步二等奖 1 项，已经发表论文 50 余篇.

《现代数学基础丛书》序

对于数学研究与培养青年数学人才而言，书籍与期刊起着特殊重要的作用．许多成就卓越的数学家在青年时代都曾钻研或参考过一些优秀书籍，从中汲取营养，获得教益．

20 世纪 70 年代后期，我国的数学研究与数学书刊的出版由于文化大革命的浩劫已经破坏与中断了 10 余年，而在这期间国际上数学研究却在迅猛地发展着．1978 年以后，我国青年学子重新获得了学习、钻研与深造的机会．当时他们的参考书籍大多还是50年代甚至更早期的著述．据此，科学出版社陆续推出了多套数学丛书，其中《纯粹数学与应用数学专著》丛书与《现代数学基础丛书》更为突出，前者出版约 40 卷，后者则逾 80 卷．它们质量甚高，影响颇大，对我国数学研究、交流与人才培养发挥了显著效用．

《现代数学基础丛书》的宗旨是面向大学数学专业的高年级学生、研究生以及青年学者，针对一些重要的数学领域与研究方向，作较系统的介绍．既注意该领域的基础知识，又反映其新发展，力求深入浅出，简明扼要，注重创新．

近年来，数学在各门科学、高新技术、经济、管理等方面取得了更加广泛与深入的应用，还形成了一些交叉学科．我们希望这套丛书的内容由基础数学拓展到应用数学、计算数学以及数学交叉学科的各个领域．

这套丛书得到了许多数学家长期的大力支持，编辑人员也为其付出了艰辛的劳动．它获得了广大读者的喜爱．我们诚挚地希望大家更加关心与支持它的发展，使它越办越好，为我国数学研究与教育水平的进一步提高做出贡献．

杨　乐

2003 年 8 月

前　言

假如有人问道: "什么是数学?" 大概许多人首先会想到几何学, 那个在公元前三百年由欧几里得总结的, 从五条几何公理出发而推出的整个完美的几何学结构. 欧几里得开创了严密的逻辑证明的方法, 展示了一切数学命题之证明必须从定义出发, 根据公理和由公理推出的定理, 再运用正确的逻辑规则进行推理, 得到的结论才是在数学上可以接受的命题. 从那时开始, 公理化体系和严密的逻辑推理成就了数学的形象. 数学, 仿佛成了最严密、最无懈可击的科学.

20 世纪初, 由于集合论中罗素悖论等的出现, 直接冲击了古典数学基础. 于是, 希尔伯特希望建立一个包罗万象的形式化的公理化体系, 从而一劳永逸地解决数学基础问题. 但在 1931 年奥地利数学家哥德尔证明了: 对于任何一个公理化体系, 都存在 "既不能证明它对, 也不能证明它错" 的命题. 这是对公理化体系的一个冲击, 但它并没有动摇大多数人对严格数学证明的信心. 毕竟, 只要接受少数公理, 即被大众直觉承认的真理, 能够证出来的东西就应该认为是对的.

然而, 近年来对公理化的证明的怀疑却是由于对一些过于复杂的证明的不信任而引起的. 例如怀尔斯证明费马大定理, 文章长达 200 多页. 据说, 全世界看得懂的不超过 20 人. 于是有人质疑: "费马大定理的证明是不是一种正在消逝的文化的最后挣扎呢? 怀尔斯是一位杰出的遗少吗?" 还有的数学家认为: "把数学在原则上简化为形式证明是 20 世纪所特有的一个不可靠的念头, 高度形式化的证明比那些借助更直观的证明更有可能出毛病."[10]

与可靠性相比, 更重要的是这种证明的有效性, 有科学家指出: "背离传统的证明的潮流或许是不可避免的. 单靠人的思维无法证明的东西是一片汪洋大海, 与这片大海比起来, 你能证明的东西, 或许只是一些孤零零的小岛, 一些例外情况而已."[10]

另一方面, 为了解决实际问题, 特别是当代高科技中涌现出的许多数学问题, 逼得人们不得不去面对那一片汪洋大海. 计算机的出现为解决这类问题提供了有力工具. 众所周知, 甚至连一些看似简单的常微分方程, 也常常无法得到闭式解. 而有限元方法却对较之复杂得多的偏微分方程几乎无所不能. 在计算机的帮助下, 数值方法成了对付那片汪洋大海的有力工具. 即使对纯粹数学问题, 一个有趣的例子是: 当 1852 年提出的四色问题让许多最杰出的数学家们痛苦了一百多年后, 在 1976 年由计算机证明了. 此后, 在复杂系统等的研究中, 人工生命、多自主体涌现等都在计算机中首先得到发现或验证. 最近看了一个研究报告《面向 2020 年的科学》[73],

报告称: “从计算机支持科学家做传统的科学研究转变为计算机科学嵌入到科学的具体结构和从事科研的方式中, 这一转变将会是一项意义重大的根本性变革.”

自从 17 世纪后期牛顿–莱布尼茨发明微积分开始, 以微积分为代表的连续性数学在自然科学的研究中起着重大作用, 在数学中占有统治地位. 计算机的出现和计算机科学的发展, 正在动摇微积分的统治地位. 计算机真正能处理的是有限值的情况, 它凸显了离散型数学的重要性. 实际上, 自然界的演化过程大致有两类: 一类是基于连续变量的动力学过程, 如行星运动、机器人行为、化学反应等, 它们可以用以微积分为代表的连续数学工具, 如微分方程等来描述和研究. 人类在这个方向上的研究成果已经很多, 数学工具也很丰富. 另一类是逻辑过程, 如布尔网络、博弈或决策过程, 它们通常只能用逻辑或离散量来描述. 随着科学技术的进步, 后者似乎正在变得越来越重要. 因此, 许多人相信: 在计算机时代, 数值化的离散型数学将会取代微积分的统治地位.

文献 [73] 中提到 “关键性新概念工具 (如微积分) 或技术工具 (如望远镜、电子显微镜) 的发明, 构成了曾经在历史和社会进程的科学革命基石的典型代表. 此类概念和技术工具现在正出现在计算机科学、数学、生物学、化学和工程学的交叉领域.” 作者相信, 数值化的离散型数学正是这种在交叉领域不断成长和逐渐成熟的一种新的概念和技术工具.

计算机从本质上说只能处理离散的、有限个数值的情况. 因此, 当处理连续性问题时就必须首先进行离散化处理. 当对象本身就是离散或有限值时, 或直接从离散或有限的对象出发, 研究其静态的映射规则与动态的演化规律, 得到的规律或许就会逐步形成这种新型的数学. 本书的目的, 就是要以有限集合为对象, 考虑有限集合上的映射的表达及其性质, 有限集合上的动态系统的演化规律及控制. 研究的主要工具是作者首创的矩阵的半张量积.

矩阵的半张量积最初目标是处理高维数组的运算. 处理高维数组的想法来自计算机内存, 即靠运算法则自动寻找表示数组层次的指针, 指针的指针……它的一个重要目标是要解决多线性及非线性问题在计算机上的运算 —— 基于矩阵形式, 而它的任何一个有意义的例子和应用都必须在计算机上验证或实现. 也许, 这就是它难以在缺少计算机的时代出现的原因. 随着科学和技术的发展, 多线性及非线性成为科学研究及技术开发中亟待解决的关键问题. 半张量积的内涵是多线性映射的矩阵化, 它的难点是计算的复杂性, 因此, 它为使用计算机解决非线性问题提供了有力工具. 如果把矩阵半张量积理论称为计算机时代的矩阵理论或矩阵算法, 大概是有道理的.

推出矩阵半张量积的过程是艰难的, 开始, 几乎没有人相信在矩阵乘法这样一个初等概念上会有什么有意义的突破. 在多次失败的尝试后, 第一作者转向国内并将基本概念结合应用问题推出. 第一篇关于矩阵半张量积的文章是结合控制系统的

Morgan 问题数值解而提出的[39]. 接着, 半张量积的主体结构在文献 [40] 中给出. 此后, 结合电力系统稳定域、非线性控制设计, 以及其他数学物理等问题, 矩阵半张积理论与方法得到应用和发展[6, 7, 41-45, 47-51]. 这些工作主要研究的都是连续系统. 在这些工作的基础上, 作者出版了一本专著[1], 它成为"现代数学基础丛书"中的一本. 清华大学梅生伟教授等继续在半张量积的电力系统应用方面工作, 得到一系列新进展. 有关工作可在他们的专著[8] 中找到.

在文献 [1] 中有一章, 是关于逻辑的半张量表示. 这个工作的初衷是一种纯数学的兴趣. 2008 年元月, 第一作者在香港第三次中瑞双边控制会议上听到清华大学赵千川教授关于布尔网络的报告. 当即感到矩阵半张量积, 特别是逻辑的矩阵表示, 可能成为分析布尔网络的拓扑结构的有效工具. 此后两年多, 半张量积在布尔网络的研究中取得成功, 初步形成了确定型布尔网络控制理论的完整框架, 相应结果发表于多篇相关国际期刊论文, 并形成专著[55]. 这方面的工作得到国际同行很高的评价, 文献 [52] 获得国际自动控制联合会 (IFAC) 颁发的 *Automatica* (三年一篇) 最佳理论/方法论文奖. 同时, 引发了一系列后继研究, 例如[74-76, 86, 93-96, 98, 99, 143, 146, 148] 等. 文献 [3] 对半张量积进行了较详尽的介绍, 文献 [60] 概述了半张量积的一些新进展.

矩阵半张量积在多方面的应用, 促使作者去思考这样一个问题: 什么是矩阵半张量积的本质, 它的优势和它所能处理的问题? 思考的结论是: 矩阵半张量积为刻画有限多个有限集之间的相互关系提供了一个清晰而又便于计算的方式. 因此, 它可以方便地应用于有限集间的映射和有限集上的演化系统的研究. 这就是本书的出发点, 它以矩阵半张量积为工具, 重新审视一系列重要的有限集间的映射及其性质; 有限集上的演化系统的动态规律及控制. 本书也会涉及多线性问题. 多线性映射虽然涉及连续集, 但它们完全由其基底性质决定. 因此, 只要对有限基底的性质弄清楚了, 问题也就解决了. 这就是半张量积对多线性映射的研究十分有效的原因.

全书共 21 章, 分为三部分. 第一部分由 1-8 章组成, 讨论矩阵半张量集与有限集映射. 第 1 章简单介绍矩阵半张量积及其主要性质; 第 2 章讲述有限集上的映射的代数表示; 第 3-6 章分别讨论布尔函数、布尔多项式、布尔代数、布尔矩阵; 第 7 章讨论逻辑函数的分解并介绍隐函数存在定理; 第 8 章讨论布尔微积分. 第二部分由 9-14 章组成, 讨论有限集上的动态系统. 第 9 章讨论布尔 (控制) 网络的拓扑结构; 第 10 章讨论布尔 (控制) 网络的状态空间及子空间; 第 11-14 章分别讨论逻辑控制系统的能控与能观性、干扰解耦、稳定性与镇定以及逻辑系统的辨识. 第三部分由 15-21 章组成, 讨论有限博弈. 第 15 章介绍非合作博弈的一般概念与初步分析; 第 16 章给出演化博弈的建模与分析; 第 17 章给出势博弈的检验方法与势函数计算; 第 18 章讨论有限博弈的向量空间结构与空间分解; 第 19 章介绍对称博弈; 第 20 章讨论合作博弈的基本概念, 以及包括 Shapley 值在内的各种分配的性质与

计算; 第 21 章讨论从一般逻辑控制系统到演化博弈的优化控制.

本书的大部分结果, 是作者近年的工作. 它们代表了在矩阵半张量积的帮助下, 人们对逻辑系统与逻辑动态系统的新探索. 许多结果带有方向性和启发性, 有待进一步的深入研究. 作者期待更多年轻人的加入, 共同开拓这块人们刚开始涉足的处女地.

本书所需要的预备知识仅为大学工科的微积分和线性代数. 部分内容涉及控制论, 所以如果有一点线性系统理论及初等概率论的知识就更好了. 但本书力求做到自给自足, 即使不是自动控制专业的读者也应该能理解相关内容. 不感兴趣者可以跳过这些内容, 并不影响其他部分的阅读.

本书的出版得到 “十二五” 国家重点图书出版规划项目和国家自然科学基金 (项目批准号：61333001, 61273013, 61104065, 61374168) 的支持，特在此致谢.

华南理工大学乔宇鹏副教授编写了第 6 章部分内容, 并审阅了部分书稿, 亦表感谢.

作者才疏学浅, 疏漏错误难免, 敬请读者以及有关专家不吝赐教.

程代展　齐洪胜　贺风华

2015 年 6 月

符号说明

$\mathbb{C}$	复数域
$\mathbb{R}$	实数域
$\mathbb{Q}$	有理数域
$\mathbb{Z}$	整数环
$\mathbb{N}$	自然数集
$\mathbb{Z}_n$	模 n 整数 (环或域)
const.	定常数
$\mathrm{lcm}\{n,p\}$	n 与 p 的最小公倍数
$\mathcal{M}_{m\times n}$	$m\times n$ 矩阵集合
$\mathcal{M}_n$	$n\times n$ 矩阵集合
I_n	n 阶单位阵
$\mathbf{1}_n$	$(\underbrace{1,\ \cdots,\ 1}_{n})^{\mathrm{T}}$
$\mathbf{0}_n$	$(\underbrace{0,\ \cdots,\ 0}_{n})^{\mathrm{T}}$
$\mathrm{Row}_i(A)$	矩阵 A 的第 i 行
$\mathrm{Row}(A)$	矩阵 A 的行集合
$\mathrm{Col}_j(A)$	矩阵 A 的第 j 列
$\mathrm{Col}(A)$	矩阵 A 的列集合
$\mathrm{Re}\,\sigma(A)$	矩阵 A 的特征值的实部
$\mathrm{Trace}(A)$	矩阵 A 的迹
$id(i_1,\cdots,i_k;n_1,\cdots,n_k)$	k 重指标
$\otimes$	矩阵的 Kronecker 积 (张量积)
$\circ$	矩阵的 Hadamard 积
$*$	矩阵的 Khatri-Rao 积
$\ltimes$	矩阵的左半张量积
$W_{[m,n]}$	$mn\times mn$ 换位矩阵
$W_{[n]}$	$n^2\times n^2$ 换位矩阵
$\mathcal{D}_k$	$\{1,2,\cdots,k\}$
$\mathcal{D}$	$\{0,1\}$

δ_k^i	I_k 的第 i 列
Δ_k	$\{\delta_k^i \mid 1 \leqslant i \leqslant k\}$
$M_{r,k}$	k 维降阶矩阵
$\wedge$	合取 (最小门)
$\vee$	析取 (最大门)
$\neg$	非
$\rightarrow$	蕴含
$\leftrightarrow$	等值
$\bar{\vee}$	异或
$\uparrow$	与非
$\downarrow$	或非
$\mathcal{L}_{m\times n}$	$m \times n$ 逻辑矩阵集合
Υ_k	k 维随机向量集合
$\Upsilon_{m\times n}$	$m \times n$ 随机矩阵集合
$\mathcal{F}_\ell\{x_1, \cdots, x_n\}$	逻辑系统的状态空间
$\mathcal{I}(F)$	逻辑映射 F 的关联矩阵
E_i	投影矩阵
$\mathcal{B}_n$	n 维布尔向量集合
$\mathcal{B}_{m\times n}$	$m \times n$ 维布尔矩阵集合
$\mathcal{B}_n^{\mathcal{F}}$	n 元布尔函数集合
$+_{\mathcal{B}}$	布尔加
$\times_{\mathcal{B}}$	布尔矩阵积
$\sum_{\mathcal{B}\ i=1}^{n}$	布尔连加
$\dashv_{[i]}$	布尔向量的 i 前置和
$A^{(k)}$	布尔 k 次幂
$w_H(X)$	布尔向量 X 的汉明权重
$d_H(X,Y)$	布尔向量 X, Y 的汉明距离
$w_b(A)$	布尔矩阵 A 的权
$D_v(X,Y)$	布尔矩阵 X 与 Y 的向量距离
$R_T(x_0)$	从 x_0 出发, T 步可达集合
$R(x_0)$	从 x_0 出发的可达集合
$A\Delta B$	集合 A, B 的对称差
$\mathrm{conv}(S)$	集合 S 的凸包
$\mathrm{argmax}_i(f(i))$	使 f 取最大的自变量集合

$\mathrm{Hess}(h(x))$	$h(x)$ 的 Hesse 矩阵
$V_1 \uplus V_2$	两个向量空间的直和
$V_1 \oplus V_2$	两个向量空间的正交和
$H < G$	H 是 G 的子群
$H \triangleleft G$	H 是 G 的正规子群
$\boldsymbol{S}_n$	n 阶对称群
$\mathrm{Fix}(g)$	群元素 g 的不动点集
BR	最佳响应策略集合

目　录

第 1 章　矩阵半张量积

1.1　高 阶 数 组

如非另加说明, 本章讨论的数组均为有限集. 一个数组, 如果必须用 k 个指标来标注它, 则称它为 k 阶数组. (k 阶数组以前也称 k 维数组, 但"维"可能与向量或矩阵的维数混淆.)

例 1.1.1　1) $\{a_i|i=1,2,\cdots,n\}$ 是一阶数组, 一阶数组可用一个向量表示. 即

$$V=(a_1\ a_2\ \cdots\ a_n)^{\mathrm{T}}.$$

2) $\{a_{i,j}|i=1,\cdots,m;j=1,\cdots,n\}$ 是二阶数组, 二阶数组可用一个矩阵表示. 即

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

3) $\{a^k_{i,j}|i=1,\cdots,m;j=1,\cdots,n;k=1,\cdots,\ell\}$ 是三阶数组.

传统矩阵理论是处理一阶或二阶数组的有力工具, 矩阵方法几乎在自然科学所有领域里都得到广泛应用. 但美中不足的是, 对于三阶或更高阶的数组, 它却无能为力. 为了处理三阶数组, 有人建议使用立体阵 (图 1.1.1)[21]. 这种方法也得到了一些有用的结果[139].

但是, 立体阵有一些明显的弱点. 首先, 需要发展一套立体阵乘矩阵、立体阵乘向量, 以及立体阵乘立体阵的计算公式, 包括运算性质的公式. 这些, 在记忆和使用上均不方便. 其次, 对于四阶或更高阶的数组, 又能怎么办呢?

矩阵半张量积是从另一个角度设法解决该问题, 它建议将任何有限阶数组均排为直线 (向量形式) 或平面 (矩阵形式), 然后, 通过改造矩阵乘法来实现对数组层次的判定和控制. 下面先考虑多维数组的排序问题. 给定一个数组

$$D=\{x_{i_1,\cdots,i_k}|1\leqslant i_1\leqslant n_1,\cdots,1\leqslant i_k\leqslant n_k\}. \tag{1.1.1}$$

引入一个记号, 称为多重指标, 记作 $id(i_1,\cdots,i_k;n_1,\cdots,n_k)$. 称数组 D 按指标 $id(i_1,\cdots,i_k;n_1,\cdots,n_k)$ 排序 (用 $\prec$ 表示前后顺序), 如果存在 $1\leqslant s\leqslant k$, 使得

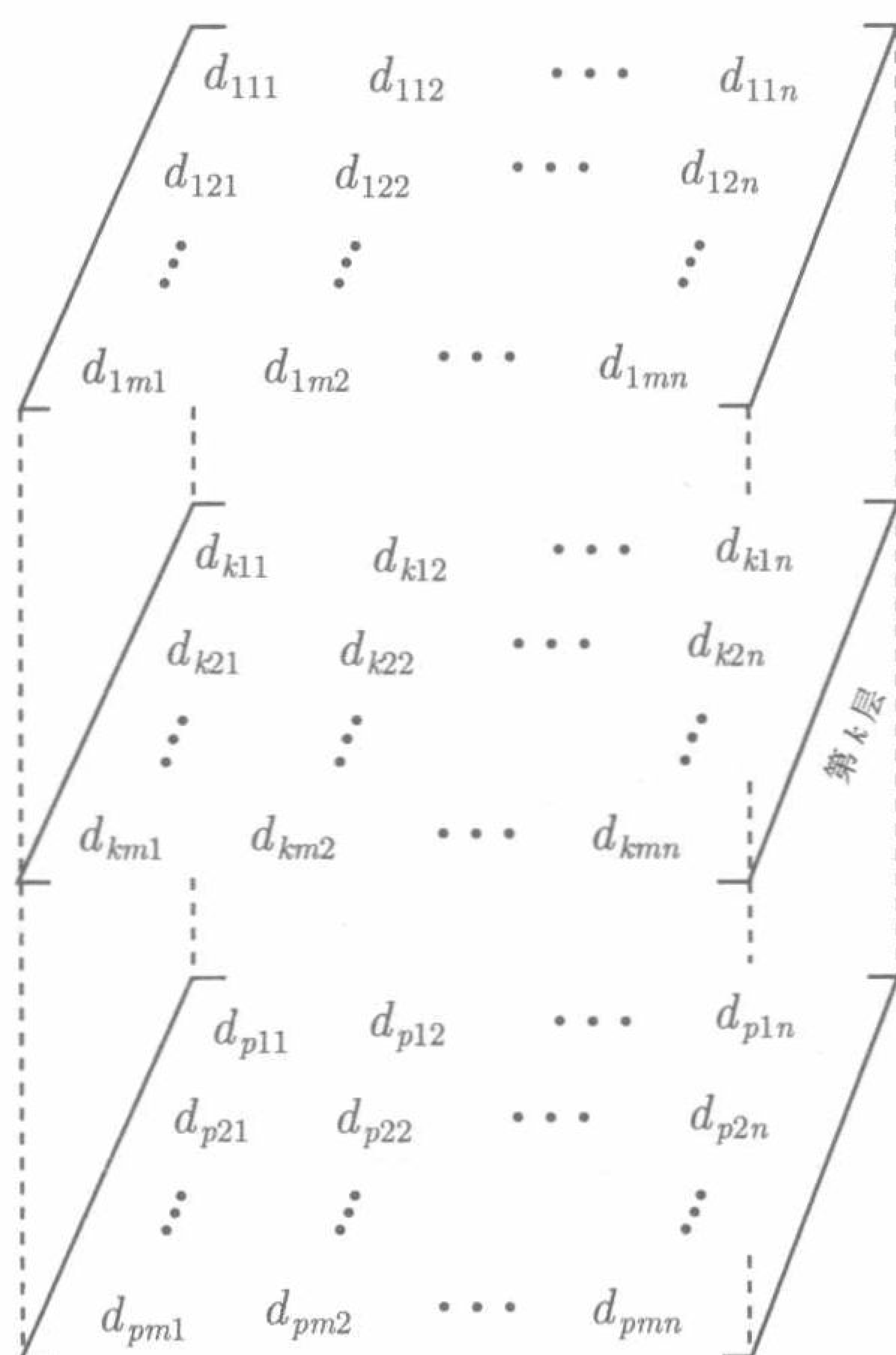


图 1.1.1　立体阵

$i_\alpha = j_\alpha$, $\alpha < s$, 且 $i_s < j_s$, 则

$$x_{i_1,\cdots,i_k} \prec x_{j_1,\cdots,j_k}.$$

例 1.1.2　设

$$D = \{x_{i,j,k} | 1 \leqslant i \leqslant 3, 1 \leqslant j \leqslant 5, 1 \leqslant k \leqslant 4\}.$$

1) 依 $id(i,j,k;3,5,4)$ 排序有

$$\begin{aligned} V_D = [&x_{111}\ x_{112}\ x_{113}\ x_{114}\ x_{121}\ x_{122}\ x_{123}\ x_{124} \\ &x_{131}\ x_{132}\ x_{133}\ x_{134}\ x_{141}\ x_{142}\ x_{143}\ x_{144} \\ &x_{151}\ x_{152}\ x_{153}\ x_{154}\ \cdots \\ &x_{311}\ x_{312}\ x_{313}\ x_{314}\ \cdots\ x_{351}\ x_{352}\ x_{353}\ x_{354}]. \end{aligned}$$

2) 依 $id(j,k,i;5,4,3)$ 排序有

$$\begin{aligned} V_D = [&x_{111}\ x_{211}\ x_{311}\ x_{112}\ x_{212}\ x_{312} \\ &x_{113}\ x_{213}\ x_{313}\ x_{114}\ x_{214}\ x_{314}\ \cdots \\ &x_{151}\ x_{251}\ x_{351}\ x_{152}\ x_{252}\ x_{352} \\ &x_{153}\ x_{253}\ x_{353}\ x_{154}\ x_{254}\ x_{354}]. \end{aligned}$$

考虑数组 (1.1.1). 设 $\{i_{r_1},i_{r_2},\cdots,i_{r_p}\}$ 和 $\{i_{s_1},i_{s_2},\cdots,i_{s_q}\}$ 为 $\{i_1,\cdots,i_k\}$ 的一个分割. 我们说数组 D 按指标 $id(i_{r_1},i_{r_2},\cdots,i_{r_p};n_{r_1},n_{r_2},\cdots,n_{r_p})\times id(i_{s_1},i_{s_2},\cdots,i_{s_q};n_{s_1},n_{s_2},\cdots,n_{s_q})$ 排成一矩阵, 如果矩阵的行依 $id(i_{r_1},i_{r_2},\cdots,i_{r_p};n_{r_1},n_{r_2},\cdots,n_{r_p})$ 排列, 而矩阵的列依 $id(i_{s_1},i_{s_2},\cdots,i_{s_q};n_{s_1},n_{s_2},\cdots,n_{s_q})$ 排列.

例 1.1.3 设

$$D=\{x_{i,j,k,r}|1\leqslant i\leqslant 2,1\leqslant j\leqslant 3,1\leqslant k\leqslant 2,1\leqslant r\leqslant 3\}.$$

1) 依 $id(i,j;2,3)\times id(k,r;2,3)$ 排序有

$$M_D=\begin{bmatrix} x_{1111} & x_{1112} & x_{1113} & x_{1121} & x_{1122} & x_{1123}\\ x_{1211} & x_{1212} & x_{1213} & x_{1221} & x_{1222} & x_{1223}\\ x_{1311} & x_{1312} & x_{1313} & x_{1321} & x_{1322} & x_{1323}\\ x_{2111} & x_{2112} & x_{2113} & x_{2121} & x_{2122} & x_{2123}\\ x_{2211} & x_{2212} & x_{2213} & x_{2221} & x_{2222} & x_{2223}\\ x_{2311} & x_{2312} & x_{2313} & x_{2321} & x_{2322} & x_{2323} \end{bmatrix}.$$

2) 依 $id(i,k;2,2)\times id(j,r;3,3)$ 排序有

$$M_D=\begin{bmatrix} x_{1111} & x_{1112} & x_{1113} & x_{1211} & x_{1212} & x_{1213} & x_{1311} & x_{1312} & x_{1313}\\ x_{1121} & x_{1122} & x_{1123} & x_{1221} & x_{1222} & x_{1223} & x_{1321} & x_{1322} & x_{1323}\\ x_{2111} & x_{2112} & x_{2113} & x_{2211} & x_{2212} & x_{2213} & x_{2311} & x_{2312} & x_{2313}\\ x_{2121} & x_{2122} & x_{2123} & x_{2221} & x_{2222} & x_{2223} & x_{2321} & x_{2322} & x_{2323} \end{bmatrix}.$$

1.2 矩阵的半张量积

先回忆一下矩阵的张量积, 张量积也称 Kronecker 积, 对任意两个矩阵 $A=(a_{ij})\in\mathcal{M}_{m\times n}$ 及 $B\in\mathcal{M}_{p\times q}$. A 与 B 的张量积定义为

$$A\otimes B:=\begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B\\ a_{21}B & a_{22}B & \cdots & a_{2n}B\\ \vdots & \vdots & & \vdots\\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix}\in\mathcal{M}_{mp\times nq}. \tag{1.2.1}$$

为方便, 将本书用到的张量积的一些主要性质罗列如下:

命题 1.2.1 1) (结合律)

$$A\otimes(B\otimes C)=(A\otimes B)\otimes C. \tag{1.2.2}$$

2) (分配律)

$$(\alpha A+\beta B)\otimes C=\alpha(A\otimes C)+\beta(B\otimes C), \tag{1.2.3}$$

$$A\otimes(\alpha B+\beta C)=\alpha(A\otimes B)+\beta(A\otimes C),\quad \alpha,\beta\in\mathbb{R}. \tag{1.2.4}$$

命题 1.2.2　1)

$$(A\otimes B)^{\mathrm{T}}=A^{\mathrm{T}}\otimes B^{\mathrm{T}}. \tag{1.2.5}$$

2) 设 A 与 B 可逆. 则

$$(A\otimes B)^{-1}=A^{-1}\otimes B^{-1}. \tag{1.2.6}$$

3)

$$\mathrm{rank}(A\otimes B)=\mathrm{rank}(A)\mathrm{rank}(B). \tag{1.2.7}$$

4) 设 $A\in\mathcal{M}_{m\times m}$, $B\in\mathcal{M}_{n\times n}$. 则

$$\det(A\otimes B)=(\det(A))^n(\det(B))^m, \tag{1.2.8}$$

$$\mathrm{Trace}(A\otimes B)=\mathrm{Trace}(A)\mathrm{Trace}(B). \tag{1.2.9}$$

下面这个性质特别有用.

命题 1.2.3　设 $A\in\mathcal{M}_{m\times n}$, $B\in\mathcal{M}_{p\times q}$, $C\in\mathcal{M}_{n\times r}$, 以及 $D\in\mathcal{M}_{q\times s}$. 那么

$$(A\otimes B)(C\otimes D)=(AC)\otimes(BD). \tag{1.2.10}$$

特别地, 有

$$A\otimes B=(A\otimes I_p)(I_n\otimes B). \tag{1.2.11}$$

定义 1.2.1　设 $A\in\mathcal{M}_{m\times n}$, $B\in\mathcal{M}_{p\times q}$, $\ell=\mathrm{lcm}\{n,p\}$ 为 n 与 p 的最小公倍数. 那么, A 与 B 的半张量积定义为

$$A\ltimes B=\left(A\otimes I_{\ell/n}\right)\left(B\otimes I_{\ell/p}\right). \tag{1.2.12}$$

注　设 $A\in\mathcal{M}_{m\times n}$, $B\in\mathcal{M}_{p\times q}$.

(i) 当 $n=p$ 时, 称 A, B 满足等维数条件, 这时半张量积与普通矩阵乘积一致. 因此, 半张量积是普通矩阵乘积的推广.

(ii) 当 $n=pt$ (或 $nt=p$), $t\in\mathbb{Z}$ 时, 称 A, B 满足倍维数条件. 记作 $A\succ_t B$ ($A\prec_t B$).

(iii) 不满足倍维数条件则称一般情况. 对一般矩阵的半张量积, 目前尚未发现有效的应用.

因为满足倍维数条件的情况特别重要, 我们对这种情况给出一个等价的定义. 它不仅有助于理解半张量积的物理意义, 而且便于计算.

定义 1.2.2 1) 设 X 为一 np 维行向量, Y 为一 p 维列向量. 将 X 分成 p 等分 $X^1,\cdots,X^p$, 这里 X^i 为 $1\times n$ 行向量. 定义 X 和 Y 的半张量积 (记作 $\ltimes$) 为

$$\begin{cases} X\ltimes Y=\sum_{i=1}^{p}X^iy_i\in\mathbb{R}^n, \\ Y^{\mathrm{T}}\ltimes X^{\mathrm{T}}=\sum_{i=1}^{p}y_i(X^i)^{\mathrm{T}}\in\mathbb{R}^n. \end{cases} \tag{1.2.13}$$

2) 设 $A\in\mathcal{M}_{m\times n}$, $B\in\mathcal{M}_{p\times q}$. $A\succ_t B$ 或 $A\prec_t B$. 那么, A 和 B 的半张量积, 记作 $C=A\ltimes B$, 定义如下: 乘积 C 由 $m\times q$ 块组成, 即 $C=(C^{ij})$, 其中, 每块定义为

$$C^{ij}=A^i\ltimes B_j,\quad i=1,\cdots,m,\quad j=1,\cdots,q,$$

这里 $A^i=\mathrm{Row}_i(A)$ 是 A 的第 i 行, $B_j=\mathrm{Col}_j(B)$ 是 B 的第 j 列.

读者请自行验证上述定义与一般定义 1.2.1 在倍维数时等价. 下面给几个数值计算的例子.

例 1.2.1 1) 设 $X=\begin{bmatrix}1 & 2 & 3 & -1\end{bmatrix}$, $Y=\begin{bmatrix}1\\2\end{bmatrix}$. 那么

$$X\ltimes Y=\begin{bmatrix}1 & 2\end{bmatrix}\cdot 1+\begin{bmatrix}3 & -1\end{bmatrix}\cdot 2=\begin{bmatrix}7 & 0\end{bmatrix}.$$

2) 设

$$A=\begin{bmatrix}1 & 2 & 1 & 1\\ 2 & 3 & 1 & 2\\ 3 & 2 & 1 & 0\end{bmatrix},\quad B=\begin{bmatrix}1 & -2\\ 2 & -1\end{bmatrix}.$$

那么

$$A\ltimes B=\begin{bmatrix}(1\ 2)\cdot 1+(1\ 1)\cdot 2 & (1\ 2)\cdot(-2)+(1\ 1)\cdot(-1)\\ (2\ 3)\cdot 1+(1\ 2)\cdot 2 & (2\ 3)\cdot(-2)+(1\ 2)\cdot(-1)\\ (3\ 2)\cdot 1+(1\ 0)\cdot 2 & (3\ 2)\cdot(-2)+(1\ 0)\cdot(-1)\end{bmatrix}=\begin{bmatrix}3 & 4 & -3 & -5\\ 4 & 7 & -5 & -8\\ 5 & 2 & -7 & -4\end{bmatrix}.$$

1.3 半张量积的性质

本节介绍一些半张量积的基本性质, 实际上, 矩阵普通乘积的性质当推广到半张量积时都得到保持. 这可能是半张量积的一个最大优点. 因为你可以像普通矩阵乘法那样进行运算, 而无须担心其维数, 只要心中记得这是半张量积即可.

命题 1.3.1　矩阵的半张量积满足

1) (分配律)

$$
\begin{aligned}
&A \ltimes (\alpha B + \beta C) = \alpha A \ltimes B + \beta A \ltimes C; \\
&(\alpha B + \beta C) \ltimes A = \alpha B \ltimes A + \beta C \ltimes A, \quad \alpha, \beta \in \mathbb{R}.
\end{aligned} \tag{1.3.1}
$$

2) (结合律)

$$
\begin{aligned}
&A \ltimes (B \ltimes C) = (A \ltimes B) \ltimes C; \\
&(B \ltimes C) \ltimes A = B \ltimes (C \ltimes A).
\end{aligned} \tag{1.3.2}
$$

命题 1.3.2　设 $A \in \mathcal{M}_{p\times q}$, $B \in \mathcal{M}_{m\times n}$. 如果 $q = km$, 那么

$$
A \ltimes B = A(B \otimes I_k); \tag{1.3.3}
$$

如果 $kq = m$, 那么

$$
A \ltimes B = (A \otimes I_k)B. \tag{1.3.4}
$$

命题 1.3.3

$$
(A \ltimes B)^{\mathrm{T}} = B^{\mathrm{T}} \ltimes A^{\mathrm{T}}; \tag{1.3.5}
$$

如果 A 及 B 是可逆阵, 那么

$$
(A \ltimes B)^{-1} = B^{-1} \ltimes A^{-1}. \tag{1.3.6}
$$

命题 1.3.4　给定 $A \in \mathcal{M}_{m\times n}$.

1) 令 $Z \in \mathbb{R}^t$ 为一行向量, 那么

$$
A \ltimes Z = Z \ltimes (I_t \otimes A); \tag{1.3.7}
$$

2) 令 $Z \in \mathbb{R}^t$ 为一列向量, 那么

$$
Z \ltimes A = (I_t \otimes A) \ltimes Z. \tag{1.3.8}
$$

命题 1.3.5　1) 设 X, Y 为两个列向量, 那么

$$
X \ltimes Y = X \otimes Y; \tag{1.3.9}
$$

2) 设 X, Y 为两个行向量, 那么

$$
X \ltimes Y = Y \otimes X. \tag{1.3.10}
$$

1.4 换位矩阵

先介绍几个记号. 记 δ_n^k 为单位阵 I_n 的第 k 列.

$$\Delta_n = \left\{\delta_n^k \middle| k = 1, 2, \cdots, n\right\}.$$

如果一个矩阵 $A \in \mathcal{M}_{n\times m}$ 的每列均只有一个 "1", 其余元素为 "0", 即 $\mathrm{Col}(A) \subset \Delta_n$, 则称 A 为一逻辑矩阵. 如果 A 是一个逻辑矩阵, 那么, 它可写成

$$A = \left[\delta_n^{k_1}\ \delta_n^{k_2}\ \cdots\ \delta_n^{k_m}\right], \quad 1 \leqslant k_i \leqslant n,\ i = 1, \cdots, m.$$

把它简记作

$$A = \delta_n\left[k_1\ k_2\ \cdots\ k_m\right].$$

定义 1.4.1 换位矩阵 $W_{[m,n]}$ 是一个 $mn \times mn$ 矩阵, 定义为

$$\begin{aligned} W_{[m,n]} = \delta_{mn}[\, & 1, m+1, 2m+1, \cdots, (n-1)m+1, \\ & 2, m+2, 2m+2, \cdots, (n-1)m+2, \\ & \cdots \\ & m, 2m, 3m, \cdots, nm]. \end{aligned} \tag{1.4.1}$$

换位矩阵可依下面的步骤构成: 将其列用 $(I, J) = (11, 12, \cdots, 1n, \cdots, m1, m2, \cdots, mn)$ 标注, 并将其行用 $(i, j) = (11, 21, \cdots, m1, \cdots, 1n, 2n, \cdots, mn)$ 标注. 那么, 它在位置 $[(i, j), (I, J)]$ 处的值为

$$w_{(i,j),(I,J)} = \delta_{i,j}^{I,J} = \begin{cases} 1, & I = i \text{ 且 } J = j, \\ 0, & \text{其他}. \end{cases} \tag{1.4.2}$$

当 $m = n$ 时, 将 $W_{[n,n]}$ 简记为 $W_{[n]}$.

例 1.4.1 设 $m = 2$ 及 $n = 3$, 那么换位矩阵 $W_{[2,3]}$ 可表示为

$$W_{[2,3]} = \begin{array}{c} \begin{matrix} (11) & (12) & (13) & (21) & (22) & (23) \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} (11) \\ (21) \\ (12) \\ (22) \\ (13) \\ (23) \end{matrix} \end{array}.$$

下面说明换位矩阵的基本作用:

设 $A \in \mathcal{M}_{m\times n}$. 记 $V_r(A)$ 为 A 的行展开, 即

$$V_r(A) = [a_{11} \cdots a_{1n} \ \cdots \ a_{m1} \cdots a_{mn}]^{\mathrm{T}},$$

记 $V_c(A)$ 为 A 的列展开, 即

$$V_c(A) = [a_{11} \cdots a_{m1} \ \cdots \ a_{1n} \cdots a_{mn}]^{\mathrm{T}}.$$

以下命题表明换位矩阵的 “换位” 作用.

命题 1.4.1　1) 设 $X \in \mathbb{R}^m$ 及 $Y \in \mathbb{R}^n$ 为两列向量, 则

$$W_{[m,n]} \ltimes X \ltimes Y = Y \ltimes X, \quad W_{[n,m]} \ltimes Y \ltimes X = X \ltimes Y. \tag{1.4.3}$$

2) 设 $X \in \mathbb{R}^m$ 及 $Y \in \mathbb{R}^n$ 为两行向量, 则

$$X \ltimes Y \ltimes W_{[m,n]} = Y \ltimes X, \quad Y \ltimes X \ltimes W_{[n,m]} = X \ltimes Y. \tag{1.4.4}$$

3) 设 $A \in \mathcal{M}_{m\times n}$. 那么

$$W_{[m,n]}V_r(A) = V_c(A), \quad W_{[n,m]}V_c(A) = V_r(A). \tag{1.4.5}$$

命题 1.4.2

$$W_{[m,n]}^{\mathrm{T}} = W_{[m,n]}^{-1} = W_{[n,m]}. \tag{1.4.6}$$

命题 1.4.3

$$W_{[pq,r]} = \left(W_{[p,r]} \otimes I_q\right)\left(I_p \otimes W_{[q,r]}\right). \tag{1.4.7}$$

式 (1.4.7) 两边取转置, 即得

$$W_{[r,pq]} = \left(I_p \otimes W_{[r,q]}\right)\left(W_{[r,p]} \otimes I_q\right). \tag{1.4.8}$$

1.5　多线性映射

前面曾经提到, 为处理高阶数组才引进矩阵半张量积的. 那么, 矩阵半张量积是如何用来处理高阶数组的呢? 这可以通过多线性映射来说明.

在 $\mathbb{R}^n$ 中取标准基底 $\Delta_n = \{\delta_n^1, \delta_n^2, \cdots, \delta_n^n\}$. 那么, $X \in \mathbb{R}^n$ 可表示成标准向量形式 $X = (x_1, \cdots, x_n)^{\mathrm{T}}$. 这种表达的实际意义是 $X = \sum_{i=1}^n x_i\delta_n^i$.

设有一线性函数 $\ell:\mathbb{R}^n\to\mathbb{R}$. 记

$$a_i=\ell(\delta_n^i),\quad i=1,\cdots,n.$$

$\{a_i\}$ 称为 ℓ 的结构常数. 将结构常数排成一行, 记作

$$V_\ell=[a_1\ a_2\ \cdots\ a_n].$$

它称为 ℓ 的结构矩阵. 于是显然有

$$\ell(X)=V_\ell X,\quad X\in\mathbb{R}^n. \tag{1.5.1}$$

因此, 线性函数可以用向量乘积表示.

再看双线性函数. 记 $\pi:\mathbb{R}^m\times\mathbb{R}^n\to\mathbb{R}$ 为一双线性函数. 记

$$a_{ij}=\pi(\delta_m^i,\delta_n^j),\quad i=1,\cdots,m;\ j=1,\cdots,n.$$

$\{a_{ij}\}$ 称为 π 的结构常数. 将结构常数排成一个矩阵, 记作

$$M_\pi=\begin{bmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\a_{21}&a_{22}&\cdots&a_{2n}\\\vdots&\vdots&&\vdots\\a_{m1}&a_{m2}&\cdots&a_{mn}\end{bmatrix}.$$

它称为 π 的结构矩阵. 容易检验

$$\pi(X,Y)=X^{\mathrm{T}}M_\pi Y,\quad X\in\mathbb{R}^m,\ Y\in\mathbb{R}^n. \tag{1.5.2}$$

因此, 双线性函数可以用矩阵与向量的乘积表示.

其实, 矩阵或线性代数的方法之所以在许多科学问题中极为有效, 其基本原理就是这种一维和二维线性函数的矩阵表示. 但如果考虑将它应用于三线性或多线性的情况, 则会遇到前面讨论过的高阶数组排列问题. 回到双线性情况. 如果把结构常数按自然顺序, 即按 $id(i,j;m,n)$ 排成一行. 即记

$$V_\pi=[a_{11}\ a_{12}\ \cdots\ a_{1n}\ \cdots\ a_{m1}\ a_{m2}\ \cdots\ a_{mn}].$$

那么, 不难证明

$$\pi(X,Y)=V_\pi\ltimes X\ltimes Y,\quad X\in\mathbb{R}^m,\ Y\in\mathbb{R}^n. \tag{1.5.3}$$

仔细将 (1.5.3) 式展开算一遍, 就会发现半张量积的优点. 当计算 $V_\pi\ltimes X$ 时, 会发现 X 的每个元会自动地去寻找它相应指标的系数. 从某种意义上讲, 它类似于计

算机中的数据结构. 在计算机存储中, 高阶数组也是排成一列. 例如, 在 C 语言中, 用指针、指针的指针等将数组的层次分出来. 可以说, 半张量积通过定义运算规则, 使不同层次的变量能自动寻找相应的数据层次, 或者说指标位置.

表达式 (1.5.3) 的优越之处在于, 它可以轻而易轻举地推广到多线性函数的情况. 以三线性为例, 设 $\xi:\mathbb{R}^m\times\mathbb{R}^n\times\mathbb{R}^s\to\mathbb{R}$. 记

$$a_{ijk}=\xi(\delta_m^i,\delta_n^j,\delta_s^k),\quad i=1,\cdots,m;\ j=1,\cdots,n;\ k=1,\cdots,s.$$

同样, 把结构常数按自然顺序, 即按 $id(i,j,k;m,n,s)$ 排成一行. 即记

$$V_\xi=[a_{111}\ a_{112}\ \cdots\ a_{11s}\ \cdots\ a_{1n1}\cdots\ a_{1ns}\ \cdots\ a_{mn1}\ \cdots\ a_{mns}].$$

不难检验

$$\xi(X,Y,Z)=V_\xi\ltimes X\ltimes Y\ltimes Z,\quad X\in\mathbb{R}^m,\ Y\in\mathbb{R}^n,\ Z\in\mathbb{R}^s. \tag{1.5.4}$$

显然, 这种表达形式轻而易举地解决了立体积想解决而难以解决的问题. 而且, 对于更一般的情况, 读者不妨自己将任意 (p 重) 线性函数的矩阵表达式写出.

作为一个初步的应用, 下面考虑多线性映射的结构矩阵. 设 $V_1,\cdots,V_k$ 分别为 $n_1,\cdots,n_k$ 维线性空间, W 为 m 维线性空间,

$$\varphi:V_1\times\cdots\times V_k\to W$$

为一 k 线性映射, 即对任意的 $1\leqslant i\leqslant k$, 设 $X_r\in V_r$, $r\neq i$, 以及 $X_i^1,X_i^2\in V_i$, $c_1,c_2\in\mathbb{R}$, 有

$$\begin{aligned}&\varphi(X_1,\cdots,X_{i-1},c_1X_i^1+c_2X_i^2,\cdots,X_k)\\&=c_1\varphi(X_1,\cdots,X_i^1,\cdots,X_k)+c_2\varphi(X_1,\cdots,X_i^2,\cdots,X_k).\end{aligned} \tag{1.5.5}$$

记 $\{e_1^i,\cdots,e_{n_i}^i\}$ 为 V_i 的一个基底, $\{d_1,\cdots,d_m\}$ 为 W 的一个基底. 并且

$$\varphi(e_{i_1}^1,\cdots,e_{i_k}^k)=\sum_{j=1}^m c_{i_1,\cdots,i_k}^j d_j;\quad i_s=1,\cdots,n_s,\ s=1,\cdots,k. \tag{1.5.6}$$

定义 1.5.1 称 $m\times(n_1\cdots n_k)$ 矩阵

$$M_\varphi=\begin{bmatrix}c_{11\cdots1}^1 & \cdots & c_{11\cdots n_k}^1 & \cdots & c_{n_1n_2\cdots n_k}^1\\ \vdots & & \vdots & & \vdots\\ c_{11\cdots1}^m & \cdots & c_{11\cdots n_k}^m & \cdots & c_{n_1n_2\cdots n_k}^m\end{bmatrix} \tag{1.5.7}$$

为 φ 的结构矩阵.

设 $X_i \in V_i$, $X_i = \sum_{j=1}^{n_i} a_j^i e_j^i$, $i = 1, \cdots, k$. 用一个向量表示 X_i, 即

$$X_i = (a_1^i, a_2^i, \cdots, a_{n_i}^i)^{\mathrm{T}}, \quad i = 1, \cdots, k.$$

那么, $\varphi(X_1, \cdots, X_k)$ 可以用下面给出的公式计算:

命题 1.5.1

$$\varphi(X_1, \cdots, X_k) = M_\varphi \ltimes X_1 \ltimes X_2 \ltimes \cdots \ltimes X_k. \tag{1.5.8}$$

例 1.5.1 $\mathbb{R}^3$ 中的叉乘定义为: 设 $X = x_1\boldsymbol{i} + x_2\boldsymbol{j} + x_3\boldsymbol{k}$, $Y = y_1\boldsymbol{i} + y_2\boldsymbol{j} + y_3\boldsymbol{k}$, 则

$$X \times Y = \det \begin{bmatrix} \boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k} \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}.$$

于是得到它的结构矩阵为

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{1.5.9}$$

设 $X = (3, 1, -1)^{\mathrm{T}}$, $Y = (1, 2, 1)^{\mathrm{T}}$. 则

$$X \times Y = M \ltimes X \ltimes Y = (3, -4, 5)^{\mathrm{T}}.$$

例 1.5.2 考虑四元数, 它是个四维线性空间, 标准基底是 $\{1, I, J, K\}$, 即

$$Q = \{a + bI + cJ + dK \,|\, a, b, c, d \in \mathbb{R}\}.$$

它的乘法运算规则是

(i) (线性性)

$$\begin{aligned}&(a_1 + b_1I + c_1J + d_1K) * (a_2 + b_2I + c_2J + d_2K)\\ =&a_1a_2 + a_1b_2I + a_1c_2J + a_1d_2K + b_1a_2I + b_1b_2I^2 + b_1c_2I * J\\ &+b_1d_2I * K + c_1a_2J + c_1b_2J * I + c_1c_2J^2 + c_1d_2J * K + d_1a_2K\\ &+d_1b_2K * I + d_1c_2K * J + d_1d_2K^2.\end{aligned}$$

(ii) (基底乘积)

$$\begin{aligned}&I * J = K; \quad J * K = I; \quad K * I = J;\\ &J * I = -K; \quad K * J = -I; \quad I * K = -J;\\ &I^2 = J^2 = K^2 = -1.\end{aligned}$$

容易算出, 它的结构矩阵是

$$M_Q=\begin{bmatrix}1&0&0&0&0&-1&0&0&0&0&-1&0&0&0&0&-1\\0&1&0&0&1&0&0&0&0&0&0&1&0&0&-1&0\\0&0&1&0&0&0&0&-1&1&0&0&0&0&1&0&0\\0&0&0&1&0&0&1&0&0&-1&0&0&1&0&0&0\end{bmatrix}. \tag{1.5.10}$$

结构矩阵给出一个计算四元数逆元的简单方法: $X=\begin{bmatrix}a&b&c&d\end{bmatrix}^{\mathrm{T}}\neq 0$, 有

$$M_QX=\begin{bmatrix}a&-b&-c&-d\\b&a&-d&c\\c&d&a&-b\\d&-c&b&a\end{bmatrix},$$

现在 X 的逆满足

$$X\times(X^{-1})=M_Q\ltimes X\ltimes(X^{-1})=(1,0,0,0)^{\mathrm{T}}.$$

记

$$\begin{aligned}E&:=\det(M_QX)\\&=a^4+b^4+c^4+d^4+2(a^2b^2+a^2c^2+a^2d^2+b^2c^2+b^2d^2+c^2d^2)\\&=(a^2+b^2+c^2+d^2)^2>0.\end{aligned}\tag{1.5.11}$$

因此

$$X^{-1}=(M_QX)^{-1}\begin{bmatrix}1\\0\\0\\0\end{bmatrix}:=\frac{1}{E}\begin{bmatrix}\alpha\\\beta\\\gamma\\\delta\end{bmatrix},$$

其中

$$\alpha=\det\left(\begin{bmatrix}a&-d&c\\d&a&-b\\-c&b&a\end{bmatrix}\right)=a^3+a(b^2+c^2+d^2);$$

$$\beta=-\det\left(\begin{bmatrix}b&-d&c\\c&a&-b\\d&b&a\end{bmatrix}\right)=-b^3-b(a^2+c^2+d^2);$$

$$\gamma = \det\left(\begin{bmatrix} b & a & c \\ c & d & -b \\ d & -c & a \end{bmatrix}\right) = -c^3 - c(a^2 + b^2 + d^2);$$

$$\delta = -\det\left(\begin{bmatrix} b & a & -d \\ c & d & a \\ d & -c & b \end{bmatrix}\right) = -d^3 - d(a^2 + b^2 + c^2).$$

容易检验 $X^{-1} \times X = 1$.

1.6 注释与参考

矩阵半张量积的概念最早是在 2001 年正式发表[39] 的. 早期的应用主要是针对连续动态系统. 清华大学梅生伟教授等将其用于电力系统, 得到一系列创新性的成果[8].

从 2008 年开始, 矩阵半张量积被应用于布尔网络控制, 效果显著, 主要成果见专著 [55]. 这个结果后来被推广应用于图论、模糊控制、布尔函数分布、故障检测等许多领域, 部分结果总结于文献 [60]. 这些工作也得到众多国内外的响应, 从而也促使作者去思考一个一般性的问题: “半张量积究竟能处理一类什么样的问题?” 本书可以认为是这种思考的结果: 半张量积是处理有限集上的映射和有限集上的动态演化系统的一个具有普适性的有力工具.

关于矩阵半张量积更多的讨论及本章定理的证明可见文献 [1].

第 2 章　有限集映射的代数表示

2.1　有限集的向量表示

一个具有 k 个元素的有限集记作

$$\mathcal{D}_k = \{1, 2, \cdots, k\}.$$

这里, 具体的数 $1 \leqslant i \leqslant k$ 没有数量上的意义, 它只是用来表示集合中的第 i 个元素.

例如, 在经典逻辑中, 逻辑变量可以取两个值: T (真), F (假). 通常把 T 对应为 1, F 对应为 0. 在这里, 为了统一起见, 不妨把 F 对应为"2". 这没有什么本质不同.

又如, 三值逻辑中, 一个逻辑变量 x 可取值于 $L_3 = \{0, 0.5, 1\}$. 不妨作如下对应

$$0 \sim 1, \quad 0.5 \sim 2, \quad 1 \sim 3,$$

或者

$$1 \sim 1, \quad 0.5 \sim 2, \quad 0 \sim 3.$$

如果不嫌别扭, 甚至可以定义

$$0.5 \sim 1, \quad 1 \sim 2, \quad 0 \sim 3.$$

但是, 对应一旦确定, $\mathcal{D}_k$ 中元素间的关系也就由相应的对应确定了.

后面可以看到, 离散值的变量及其相关映射在表示和数学综合上都有许多不便之处. 因为经典的、以连续变量为对象的数学工具, 如微积分、微分方程等, 很难用到这类系统上. 本书的核心思想, 就是将离散变量用向量形式表示, 结合矩阵半张量积, 使许多传统数学工具可以应用于此类系统.

我们让 $\mathcal{D}_k$ 中的元素与 Δ_k 中的元素一一对应, 即

$$i \sim \delta_k^i, \quad i = 1, 2, \cdots, k. \tag{2.1.1}$$

这个表达有一个天生的优点: 即它对乘积 (笛卡儿积) 集合的天然对应. 设 $A = \{a_1, \cdots, a_m\}$ 和 $B = \{b_1, \cdots, b_n\}$ 为两个有限集. 用 δ_m^i 代表 a_i, 用 δ_n^j 代表 b_j. 现在考虑乘积集合

$$A \times B = \{(a_i, b_j) \big| 1 \leqslant i \leqslant m,\ 1 \leqslant j \leqslant n\}.$$

$A\times B$ 中的元素可用自然排序 $id(i,j;m,n)$. 那么, $A\times B$ 中的元素可以用 $\{\delta_{mn}^i|1\leqslant i\leqslant mn\}$ 来表示. 现在, 设 $x=\delta_m^i\sim a_i$ 及 $y=\delta_n^j\sim b_j$, 那么 (x,y) 的向量表示是什么呢?

命题 2.1.1 设 $x=\delta_m^i\sim a_i$ 及 $y=\delta_n^j\sim b_j$. 那么, 依自然排序

$$(x,y)\sim x\ltimes y. \tag{2.1.2}$$

这个命题的证明很简单, 读者自己动手写一写就清楚了. 这个结论对有限个集合都成立, 即, 设 $x_k\in\Delta_{n_k}$, $k=1,\cdots,s$. 那么 $(x_1,\cdots,x_s)$ 在乘积空间自然序下的表示为

$$(x_1,x_2,\cdots,x_s)\sim\ltimes_{i=1}^s x_i. \tag{2.1.3}$$

例 2.1.1 设 $x=\delta_5^3$, 即 x 代表一个 5 元素集的第三个元素, 设 $y=\delta_6^2$, $z=\delta_8^5$. 那么 (x,y,z) 代表乘积空间中第几个元素呢? 因为

$$\delta_5^3\ltimes\delta_6^2\ltimes\delta_8^5=\delta_{240}^{109},$$

因此, 在自然排序下, (x,y,z) 是第 109 个元素.

在以后的讨论中, 我们就直接将一个元素等同于它的向量表示. 于是, 如果 $x_i=\delta_{n_i}^{k_i}$, $i=1,\cdots,s$. 那么, 它们的乘积

$$x=\ltimes_{i=1}^s\delta_{n_i}^{k_i}$$

就是它们的笛卡儿积在乘积空间中的元.

这个结论非常重要, 因为将来无论考虑映射还是动态演化, 都会在乘积空间展开. 最后, 根据唯一性, 再从乘积空间返回各因子空间. 下面讨论如何从乘积空间返回到因子空间. 为叙述方便, 定义几个记号:

1) $\mathbf{1}$ 向量:

$$\mathbf{1}_q:=(\underbrace{1,\ \cdots,\ 1}_{q})^{\mathrm{T}}.$$

2) 设有一个 n 阶数组, 其维数分别为 $k_1,\cdots,k_n$. 定义维数的部分乘积为

$$k^{[i,j]}:=\begin{cases}\prod_{s=i}^j k_s, & i\leqslant j,\\ 1, & i>j.\end{cases} \tag{2.1.4}$$

3) 定义一组投影矩阵 E_i, $i=1,\cdots,n$ 如下:

$$E_i:=\mathbf{1}_{k^{[1,i-1]}}^{\mathrm{T}}\otimes I_{k_i}\otimes\mathbf{1}_{k^{[i+1,n]}}^{\mathrm{T}},\quad i=1,\cdots,n. \tag{2.1.5}$$

命题 2.1.2 设 $x_i \in \Delta_{k_i}$, $i = 1, \cdots, n$. 那么, $x := \ltimes_{i=1}^n x_i \in \Delta_k$, 这里, $k = \prod_{i=1}^n k_i$. 而且

$$x_i = E_i x, \quad 1 \leqslant i \leqslant n. \tag{2.1.6}$$

为证明这个命题, 先给一个引理.

引理 2.1.1 ([13]) 设 $A \in \mathcal{M}_{m\times n}$, $B \in \mathcal{M}_{p\times q}$, $C \in \mathcal{M}_{n\times r}$, $D \in \mathcal{M}_{q\times s}$, 则

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD). \tag{2.1.7}$$

下面证明命题 2.1.2:

证明 不妨将 E_i 改写成

$$E_i = \otimes_{s=1}^{i-1} \mathbf{1}_{k_s}^{\mathrm{T}} \otimes I_{k_i} \otimes_{s=i+1}^{n} \mathbf{1}_{k_s}^{\mathrm{T}}. \tag{2.1.8}$$

由 (1.3.9) 可得

$$x = \ltimes_{i=1}^n x_i = \otimes_{i=1}^n x_i. \tag{2.1.9}$$

利用引理 2.1.1, 将 (2.1.8) 与 (2.1.9) 逐个对应因子相乘即得. □

我们把 $x_1, \cdots, x_s$ 称为 x 的乘积分量. 其实, 可以更抽象地定义乘积分量: 设 $x \in \Delta_k$,

$$k = k_1 k_2 \cdots k_r, \tag{2.1.10}$$

这里 $k_i \geqslant 2$. 那么, 根据 (2.1.5) 可以定义投影矩阵 $E_i, i = 1, \cdots, r$. 令

$$x_i = E_i x, \quad i = 1, \cdots, r. \tag{2.1.11}$$

则有

命题 2.1.3 1) $x_i \in \Delta_{k_i}$, $i = 1, \cdots, r$.

2)

$$x = \ltimes_{i=1}^r x_i. \tag{2.1.12}$$

证明 只要证明 $k = k_1 k_2$ 的情况就够了, 多个因子情况可递推得出. 将 x 等分为 k_1 个 k_2 维向量, $z_1, \cdots, z_{k_1}$, 其中只有一个, 设为 z_i, 含分量 1. 设 $z_i = \delta_{k_2}^j$, 则有

$$x = \delta_{k_1}^i \ltimes \delta_{k_2}^j.$$

□

注 1) 命题 2.1.3 说明存在一组 $x_i \in \Delta_{k_i}, i=1,\cdots,r$, 使得

$$x = \ltimes_{i=1}^{r} x_i.$$

容易证明, 这组因子是唯一的.

2) 当 (2.1.10) 为素因数分解时, 相应的 $\{x_i | i=1,\cdots,r\}$ 称为 x 的素乘积分量.

例 2.1.2 设 $x=\delta_{30}^7$. 作 30 的素分解: $30=2\cdot 3\cdot 5$. 据此构造

$$\begin{aligned} E_1 &= I_2 \otimes \mathbf{1}_{15}^{\mathrm{T}}, \\ E_2 &= \mathbf{1}_2^{\mathrm{T}} \otimes I_3 \otimes \mathbf{1}_5^{\mathrm{T}}, \\ E_3 &= \mathbf{1}_6^{\mathrm{T}} \otimes I_5. \end{aligned}$$

则有

$$\begin{aligned} x_1 &= E_1 x = \delta_2^1, \\ x_2 &= E_2 x = \delta_3^2, \\ x_3 &= E_3 x = \delta_5^2. \end{aligned}$$

同样可知, 当 $x=\delta_{30}^{19}$ 时,

$$x_1=\delta_2^2, \quad x_2=\delta_3^1, \quad x_3=\delta_5^4;$$

当 $x=\delta_{30}^{26}$ 时,

$$x_1=\delta_2^2, \quad x_2=\delta_3^3, \quad x_3=\delta_5^1.$$

2.2 有限集上的函数

在第 1 章介绍过逻辑矩阵: 设 $L\in\mathcal{M}_{m\times n}$. 如果 L 的每一列有且仅有一个 1, 其余为 0, 即 $\mathrm{Col}(L)\subset\Delta_m$, 则称 L 为一逻辑矩阵. $m\times n$ 维的逻辑矩阵集合记作 $\mathcal{L}_{m\times n}$.

定义 2.2.1 一个映射 $f:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_{k_0}$ 称为一个多值逻辑函数. 如果 $k_0=k_1=\cdots=k_n:=k$, 则称 f 为 k 值逻辑函数. 一个 k 值逻辑函数, 如果 $k=2$, 则称为布尔函数 (或逻辑函数).

下面使用向量表示有限集 $\mathcal{D}_k$ 中的元, 即将 $\mathcal{D}_k$ 与 Δ_k 等价. 考虑多值逻辑函数 $f:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_{k_0}$, 在向量形式下, f 变为

$$f:\Delta_{k_1}\times\Delta_{k_2}\times\cdots\times\Delta_{k_n}\to\Delta_{k_0}. \tag{2.2.1}$$

现在设 $x_i\in\Delta_{k_i}, i=1,\cdots,n$. 那么, $x:=\ltimes_{i=1}^n x_i\in\Delta_k$, 这里 $k=\prod_{i=1}^n k_i$. 注意到对每个 x 都有唯一的函数值 $f(x):=f(x_1,\cdots,x_n)$. 记

$$M_f=\left[f(\delta_k^1)\ f(\delta_k^2)\ \cdots\ f(\delta_k^k)\right], \tag{2.2.2}$$

那么, 不难验证 $M_f \in \mathcal{L}_{k_0 \times k}$, 并且 $f(x) = M_f x$. 我们将它叙述成一个定理.

定理 2.2.1　考虑多值逻辑函数 $f: \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{k_0}$, 则存在唯一的逻辑矩阵, $M_f \in \mathcal{L}_{k_0 \times k}$, 称为 f 的结构矩阵, 使得在向量形式下有

$$f(x_1, \cdots, x_n) = M_f \ltimes_{i=1}^n x_i. \tag{2.2.3}$$

我们给一个例子来说明.

例 2.2.1　A, B 二人玩石头-剪刀-布游戏. 基本规则是: 石头赢剪刀, 剪刀赢布, 布赢石头.

1) A 取如下策略: 如果赢了或平了, 下次不改变策略, 如果输了, 则下次取对方 (B) 的策略.

2) B 取如下策略: 如果与对方 (A) 的策略相同, 则下次不改变策略, 否则, 取两人均未取的策略.

先用向量值表示有限集中的元素: 用 δ_3^1 表示 "石头", δ_3^2 表示 "剪刀", δ_3^3 表示 "布". 用 $x(t)$ 和 $y(t)$ 分别表示 A 和 B 在 t 时刻的策略. 于是有

$$x(t+1) = f_1(x(t), y(t)); \tag{2.2.4}$$

及

$$y(t+1) = f_2(x(t), y(t)). \tag{2.2.5}$$

记 $z(t) := x(t) \ltimes y(t)$. 在向量形式下, (2.2.4) 的代数表达式为

$$x(t+1) = M_1 x(t) y(t) = M_1 z(t). \tag{2.2.6}$$

设 $x(t) = \delta_3^1$ 及 $y(t) = \delta_3^1$. 根据 A 的策略, 本次为平局, 下次策略不变. 又 $z(t) = x(t) \ltimes y(t) = \delta_9^1$. 于是

$$\mathrm{Col}_1(M_1) = \delta_3^1.$$

设 $x(t) = \delta_3^1$ 及 $y(t) = \delta_3^2$. 根据 A 的策略, 本次 A 赢, 下次策略不变. 又 $z(t) = x(t) \ltimes y(t) = \delta_9^2$. 于是

$$\mathrm{Col}_2(M_1) = \delta_3^1.$$

依次考查 A, B 两玩家的不同策略, 可知 A 下次的策略. 于是可得

$$M_1 = \delta_3[1\ 1\ 3\ 1\ 2\ 2\ 3\ 2\ 3].$$

同样, 由 A, B 两玩家的不同策略, 可知 B 下次的策略. 可得

$$M_2 = \delta_3[1\ 3\ 2\ 3\ 2\ 1\ 2\ 1\ 3].$$

由于一个多值逻辑函数 $f: \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{k_0}$ 与一个逻辑矩阵 $M_f \in \mathcal{L}_{k_0 \times k_1 k_2 \cdots k_n}$ 一一对应, 不难知道, 这样的 n 元多值逻辑函数的总数为

$$N = \left|\left\{f \middle| f: \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{k_0}\right\}\right| = k_0^{k_1 k_2 \cdots k_n}. \tag{2.2.7}$$

作为特殊情况, n 元 k 值逻辑函数的总数为

$$N = \left|\left\{f \middle| f: \mathcal{D}_k^n \to \mathcal{D}_k\right\}\right| = k^{k^n}. \tag{2.2.8}$$

n 元布尔函数的总数为

$$N = \left|\left\{f \middle| f: \mathcal{D}_2^n \to \mathcal{D}_2\right\}\right| = 2^{2^n}. \tag{2.2.9}$$

下面讨论布尔函数. 设布尔函数 $f: \mathcal{D}_2^n \to \mathcal{D}_2$, 这里 $\mathcal{D}_2 = \{T, F\}$. (T 代表“真”, F 代表“假”.) 一些基本的逻辑运算符定义如下:

(i) 一元逻辑算子: 非 ($\neg$),

$$\neg x = \begin{cases} F, & x = T, \\ T, & x = F. \end{cases} \tag{2.2.10}$$

(ii) 二元逻辑算子: 析取 (或) ($\vee$); 合取 (与) ($\wedge$); 蕴涵 ($\to$); 等价 ($\leftrightarrow$). 它们的逻辑值通常用一个真值表来表示 (表 2.2.1).

表 2.2.1 $\wedge, \vee, \to, \leftrightarrow$ 的真值表

p	q	$p \wedge q$	$p \vee q$	$p \to q$	$p \leftrightarrow q$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

例 2.2.2 在向量形式下, 令 $T \sim \delta_2^1$ 及 $F \sim \delta_2^2$.

1) 设 $y = \neg x$. 用 M_n 表示 $\neg$ 的结构矩阵, 即在向量形式下

$$y = M_n x,$$

则

$$M_n = \delta_2[2\ 1]. \tag{2.2.11}$$

2) 设 $z = x \vee y$. 用 M_d 表示 $\vee$ 的结构矩阵, 即在向量形式下

$$z = M_d x y,$$

则

$$M_d = \delta_2[1\ 1\ 1\ 2]. \tag{2.2.12}$$

3) 设 $z = x \wedge y$. 用 M_c 表示 $\wedge$ 的结构矩阵, 即在向量形式下

$$z = M_c xy,$$

则

$$M_c = \delta_2[1\ 2\ 2\ 2]. \tag{2.2.13}$$

4) 设 $z = x \to y$. 用 M_i 表示 $\to$ 的结构矩阵, 即在向量形式下

$$z = M_i xy,$$

则

$$M_i = \delta_2[1\ 2\ 1\ 1]. \tag{2.2.14}$$

5) 设 $z = x \leftrightarrow y$. 用 M_e 表示 $\leftrightarrow$ 的结构矩阵, 即在向量形式下

$$z = M_e xy,$$

则

$$M_e = \delta_2[1\ 2\ 2\ 1]. \tag{2.2.15}$$

2.3　映射的合成与分解

定义 2.3.1　一个映射 $F : \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{r_1} \times \mathcal{D}_{r_2} \times \cdots \times \mathcal{D}_{r_m}$ 称为一个多值逻辑映射. 如果 $k_1 = \cdots = k_n = r_1 = \cdots = r_m := k$, 则称 F 为 k 值逻辑映射. 一个 k 值逻辑映射, 如果 $k = 2$, 则称为布尔映射 (或逻辑映射).

设 $F : \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{r_1} \times \mathcal{D}_{r_2} \times \cdots \times \mathcal{D}_{r_m}$ 为一个多值逻辑映射. 那么, 不难看出, F 可由 m 个多值逻辑函数组成, 即 $F = (f_1, \cdots, f_m)$, 这里

$$f_i : \mathcal{D}_{k_1} \times \mathcal{D}_{k_2} \times \cdots \times \mathcal{D}_{k_n} \to \mathcal{D}_{r_i}, \quad i = 1, \cdots, m. \tag{2.3.1}$$

现在设 f_i 在向量形式下有代数表达式

$$\begin{aligned} f_i(x_1, \cdots, x_n) = M_i \ltimes_{j=1}^{n} x_j, \quad x_j \in \mathcal{D}_{k_j}, j = 1, \cdots, n, \\ i = 1, \cdots, m. \end{aligned} \tag{2.3.2}$$

其实, 也可以把 F 看成一个多值逻辑函数: $F:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_r$, 这里 $r=\prod_{k=1}^{m}r_k$. 因此, 多值逻辑函数与多值逻辑映射无本质区别. 现在考虑如何从各分量 f_i 的代数表达式得到 F 的代数表达式. 我们需要一种新的矩阵乘法[13].

定义 2.3.2 (Khatri-Rao 积) 设 $A\in\mathcal{M}_{p\times n}$, $B\in\mathcal{M}_{q\times n}$. 则它们的 Khatri-Rao 积, 记作 $A*B$, 定义为

$$A*B=[\mathrm{Col}_1(A)\ltimes\mathrm{Col}_1(B)\ \mathrm{Col}_2(A)\ltimes\mathrm{Col}_2(B)\ \cdots\ \mathrm{Col}_n(A)\ltimes\mathrm{Col}_n(B)]\in\mathcal{M}_{pq\times n}. \tag{2.3.3}$$

定理 2.3.1 设混合值逻辑映射 $F=(f_1,\cdots,f_m)$, 其中 f_i 具有结构矩阵 M_i, $i=1,\cdots,m$. 即

$$f_i(x_1,\cdots,x_n)=M_i\ltimes_{j=1}^{n}x_j,\quad i=1,\cdots,m.$$

则

$$F(x_1,\cdots,x_n)=M_F\ltimes_{j=1}^{n}x_j,\quad i=1,\cdots,m,$$

这里

$$M_F=M_1*M_2*\cdots*M_m. \tag{2.3.4}$$

证明 设 $x=\ltimes_{i=1}^{n}x_i=\delta_k^{\ell}$, 这里 $k=\prod_{i=1}^{n}k_i$. 那么

$$f_i(x)=\mathrm{Col}_{\ell}(M_i),\quad i=1,\cdots,m.$$

因为

$$F(x)=\ltimes_{i=1}^{m}f_i(x),$$

则

$$\mathrm{Col}_{\ell}(M_F)=\ltimes_{i=1}^{m}\mathrm{Col}_{\ell}(M_i).$$

□

现在假定 $F:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_r$ 为一逻辑映射.

$$r=r_1\times r_2\times\cdots\times r_m$$

为 r 的一组因数分解, $r_i\geqslant 2$, $i=1,\cdots,m$. 那么, 是否存在一组 $f_i:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_{r_i}$, 使得 $F=(f_1,\cdots,f_m)$ 呢? 答案是肯定的.

定理 2.3.2　设 $F:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_r$ 为一逻辑映射.

$$r=r_1\times r_2\times\cdots\times r_m$$

为 r 的一组因数分解, 那么, 存在唯一的一组 $f_i:\mathcal{D}_{k_1}\times\mathcal{D}_{k_2}\times\cdots\times\mathcal{D}_{k_n}\to\mathcal{D}_{r_i}$, 使得 $F=(f_1,\cdots,f_m)$. 而且, 设 F 的结构矩阵为 $M_F\in\mathcal{M}_{r\times k}$, 这里, $k=\prod_{i=1}^n k_i$. 则 f_i 的结构矩阵为

$$M_i=E_iM_F\in\mathcal{M}_{r_i\times k},\quad i=1,\cdots,m. \tag{2.3.5}$$

上式中 $E_i, i=1,\cdots,m$ 为投影矩阵. (见 (2.1.5).)

证明　利用命题 2.1.3 即得. □

我们称定理 2.3.1 为映射合成定理, 定理 2.3.2 为映射分解定理.

例 2.3.1　考察例 2.2.1. 我们分别得到关于玩家 A 和玩家 B 的策略更新方程:

$$\begin{aligned}x(t+1)&=M_1x(t)y(t),\\ y(t+1)&=M_2x(t)y(t).\end{aligned}$$

记 $z(t)=x(t)\ltimes y(t)$. 显然, 整个博弈的演化可以用 $z(t)$ 的更新方程来描述. 而它正好是这两个映射的合成. 由映射合成定理可知

$$z(t+1)=Mz(t),$$

这里

$$M=M_1*M_2=\delta_9[1\ 3\ 8\ 3\ 5\ 4\ 8\ 4\ 9].$$

例 2.3.2　考察一个具有如下模型的动态系统

$$x(t+1)=Mx(t), \tag{2.3.6}$$

这里 $x(t)\in\Delta_{30}$, 结构矩阵 M 为

$$\begin{aligned}M=\delta_{30}[&2,5,28,3,17,9,15,21,6,11,30,24,1,23,4,\\&17,2,23,18,7,20,1,2,26,17,13,20,19,6,8].\end{aligned}$$

1) 将其分解为 3 个分量 $x_1(t)\in\Delta_2$, $x_2(t)\in\Delta_3$, $x_3(t)\in\Delta_5$. 于是有

$$\begin{cases}x_1(t+1)=M_1x(t),\\ x_2(t+1)=M_2x(t),\\ x_3(t+1)=M_3x(t).\end{cases}$$

这里,

$$
\begin{aligned}
M_1 &= E_1 M \\
&= \delta_2[1,1,2,1,2,1,1,2,1,1,2,2,1,2,1,2,1,2,2,1,2,1,1,2,2,1,2,2,1,1]; \\
M_2 &= E_2 M \\
&= \delta_3[1,1,3,1,1,2,3,2,2,3,3,2,1,2,1,1,1,2,1,2,1,1,1,3,1,3,1,1,2,2]; \\
M_3 &= E_3 M \\
&= \delta_5[2,5,3,3,2,4,5,1,1,1,5,4,1,3,4,2,2,3,3,2,5,1,2,1,2,3,5,4,1,3].
\end{aligned}
$$

其中, E_1, E_2, E_3 可见例 2.1.2.

2) 将其分解为 2 个分量 $\tilde{x}_1(t) \in \Delta_6$, $\tilde{x}_2(t) \in \Delta_5$. 构造

$$
\tilde{E}_1 = I_6 \otimes \mathbf{1}_5^{\mathrm{T}}; \quad \tilde{E}_2 = \mathbf{1}_6^{\mathrm{T}} \otimes I_5.
$$

则可得

$$
\begin{cases}
\tilde{x}_1(t+1) = \tilde{M}_1 x(t), \\
\tilde{x}_2(t+1) = \tilde{M}_2 x(t),
\end{cases}
$$

这里

$$
\begin{aligned}
\tilde{M}_1 &= \tilde{E}_1 M \\
&= \delta_6[1,1,6,1,4,2,3,5,2,3,6,5,1,5,1,4,1,5,4,2,4,1,1,6,4,3,4,4,2,2]; \\
\tilde{M}_2 &= \tilde{E}_2 M \\
&= \delta_5[2,5,3,3,2,4,5,1,1,1,5,4,1,3,4,2,2,3,3,2,5,1,2,1,2,3,5,4,1,3].
\end{aligned}
$$

如果与前述分解相比, 不难看出: $\tilde{x}_1 = x_1 \ltimes x_2$, $\tilde{x}_2 = x_3$. 因此有

$$
\tilde{M}_1 = M_1 * M_2; \quad \tilde{M}_2 = M_3.
$$

读者可自行检验这一点.

2.4 注释与参考

从中学开始, 我们处理的函数, 如代数函数 $y = ax + b$, $y = ax^2 + bx + c$, 超越函数 $y = e^x$, 三角函数 $y = \sin(x)$ 等, 自变量都是取连续值的. 因此, 到大学的微积分, 我们可以对它们求导、积分. 还可以用微分方程表示这些取连续值的变量的动态演化过程.

但是, 另一类变量, 例如逻辑变量及其间的函数关系. 例如: $z = x \wedge y$, 这里 x, y, $z \in \mathcal{D}_2$ 都只可能取两个值: 或为 0 或为 1. 这时, 中学代数和大学微积分、微

分方程、差分方程等工具就失效了. 为了能够对这类系统使用经典的数学工具, 本章提出了一套逻辑函数的代数表示方法, 它将逻辑函数转化成一个仅涉及代数运算的形式. 到目前为止, 这个方法已经得到很大的成功, 也得到许多国内外学者的承认、接受和使用[55, 60]. 本章对已知的、散落在若干文献中的这种表示方法做了梳理和规范.

第 3 章　命题逻辑与布尔函数

3.1　布尔函数的代数表示

设 $y=f(x_1,\cdots,x_n)$ 为一布尔函数. 它的代数表达式指的是: 找出 f 的结构矩阵 M_f, 使得在变量的向量表达下, 该函数可表示为

$$y=M_f x, \tag{3.1.1}$$

这里 $x=\ltimes_{i=1}^n x_i$, $M_f\in\mathcal{M}_{2\times 2^n}$.

通常有两种办法给出它的代数表达式.

(1) 直接法:

这种方法第 2 章曾介绍过, 它对一般混合值逻辑函数都成立. 记 $x=\ltimes_{i=1}^n x_i$. 则

$$\mathrm{Col}_i(M_f)=f\left(\delta_{2^n}^i\right),\quad i=1,2,\cdots,2^n.$$

这其实可以直接从真值表得到.

例 3.1.1　设

$$w=f(x,y,z):=(x\vee y)\to(\neg z). \tag{3.1.2}$$

由真值表 (表 3.1.1) 立即可知 (取最后一列)

$$M_f=\delta_2[2\ 1\ 2\ 1\ 2\ 1\ 1\ 1].$$

表 3.1.1　(3.1.2) 的真值表

x	y	z	$x\vee y$	$\neg z$	w
T	T	T	T	F	F
T	T	F	T	T	T
T	F	T	T	F	F
T	F	F	T	T	T
F	T	T	T	F	F
F	T	F	T	T	T
F	F	T	F	F	T
F	F	F	F	T	T

(2) 间接法

间接法的基本思想是从一些基本逻辑算子已知的结构矩阵出发, 再利用矩阵半张量积的性质, 推出一般逻辑函数的结构矩阵.

例 3.1.2 考虑例 3.1.1, 利用例 2.2.2 得到的 M_i, M_d 和 M_n, 有

$$\begin{aligned} w &= M_i(M_d xy)(M_n z) \\ &= M_i M_d (xy M_n) z \\ &= M_i M_d (I_4 \otimes M_n) xyz \\ &= \delta_2[1\ 2\ 1\ 1] \ltimes \delta_2[1\ 1\ 1\ 2] \ltimes (I_4 \otimes \delta_2[2\ 1])\, xyz \\ &= M_f xyz. \end{aligned}$$

上式第一个等式来自半张量积的结合律, 第二个等式利用命题 1.3.4 中的等式 (1.3.8). 于是

$$\begin{aligned} M_f &= \delta_2[1\ 2\ 1\ 1] \ltimes \delta_2[1\ 1\ 1\ 2] \ltimes (I_4 \otimes \delta_2[2\ 1]) \\ &= \delta_2[2\ 1\ 2\ 1\ 2\ 1\ 1\ 1]. \end{aligned}$$

在使用间接法时, 有时会碰上重因子. 为对付重因子, 引入降阶矩阵.

$$M_{r,k} := \operatorname{diag}\left(\delta_k^1,\ \delta_k^2, \cdots,\ \delta_k^k\right)$$

称为 k 维降阶矩阵. $k=2$ 时简记为 M_r.

命题 3.1.1 设 $x \in \Delta_k$, 则

$$x^2 = M_{r,k} x. \tag{3.1.3}$$

直接计算即可证明上述命题. (3.1.3) 说明了降阶矩阵这一名称的来历. 下面再给一个简单例子.

例 3.1.3 考虑

$$w = f(x,y) = (x \vee y) \leftrightarrow (x \wedge y). \tag{3.1.4}$$

直接计算可得

$$\begin{aligned} w &= M_e(M_d xy)(M_c xy) \\ &= M_e M_d (I_4 \otimes M_c)(xy)^2 \\ &= M_e M_d (I_4 \otimes M_c) M_{r,4} xy \\ &= \delta_2[1\ 2\ 2\ 1]\delta_2[1\ 1\ 1\ 2]\,(I_4 \otimes \delta_2[1\ 2\ 2\ 2\ ])\,\delta_{16}[1\ 6\ 11\ 16]xy \\ &= \delta_2[1\ 2\ 2\ 1]xy. \end{aligned}$$

故

$$M_f = \delta_2[1\ 2\ 2\ 1].$$

3.2 布尔映射的代数表达

设有一布尔映射 $F : \mathcal{D}^n \to \mathcal{D}^m$, 它由下式表示

$$\begin{cases} y_1 = f_1(x_1, \cdots, x_n), \\ y_2 = f_2(x_1, \cdots, x_n), \\ \qquad \vdots \\ y_m = f_m(x_1, \cdots, x_n). \end{cases}$$

为了得到 F 的结构矩阵, 分两步走. 第一步: 给出 f_i, $i = 1, \cdots, m$ 的结构矩阵 M_i; 第二步: 从 M_i, $i = 1, \cdots, m$ 构造 M_F. 关于第一步, 3.1 节已经讨论过了. 本节只讨论第二步, 即假定 M_i 已知, 求 M_F.

方法一: 直接利用定理 2.3.1. 即

$$M_F = M_1 * M_2 * \cdots * M_m.$$

用这个方法的一个可能遇到的困难是, f_i 中可能不含某些变量 $\{x_j | j = 1, \cdots, s,$ 其中 $s < n\} \subset \{x_1, \cdots, x_n\}$. 这时需要哑矩阵. 定义

$$\begin{aligned} D_f[m, n] &:= \mathbf{1}_m^{\mathrm{T}} \otimes I_n; \\ D_r[m, n] &:= I_m \otimes \mathbf{1}_n^{\mathrm{T}}; \end{aligned}$$

利用哑矩阵, 直接计算即可得如下结果:

命题 3.2.1 设 $x \in \Delta_m$, $y \in \Delta_n$. 则

$$D_f[m, n]xy = y, \tag{3.2.1}$$

$$D_r[m, n]xy = x. \tag{3.2.2}$$

等式 (3.2.1) 及 (3.2.2) 让我们可以自由地在一个逻辑的代数表达式中增加哑变量. 用一个例子来说明.

例 3.2.1 设 $F(x, y, z)$ 由下式决定

$$\begin{cases} p = f_1(x, y, z) := x \wedge (\neg y), \\ q = f_2(x, y, z) := x \vee (y \to z). \end{cases} \tag{3.2.3}$$

构造相应的真值表, 很容易得到

$$\begin{aligned} p &= \delta_2[2\ 1\ 2\ 2]xy; \\ q &= \delta_2[1\ 1\ 1\ 1\ 1\ 2\ 1\ 1]xyz. \end{aligned}$$

利用 (3.2.2) 可得

$$\begin{aligned} p &= \delta_2[2\ 1\ 2\ 2]D_r[4,2]xyz \\ &= \delta_2[2\ 2\ 1\ 1\ 2\ 2\ 2\ 2]xyz. \end{aligned}$$

于是

$$\begin{aligned} M_F &= \delta_2[1\ 1\ 1\ 1\ 1\ 2\ 1\ 1] * \delta_2[2\ 2\ 1\ 1\ 2\ 2\ 2\ 2] \\ &= \delta_4[2\ 2\ 1\ 1\ 2\ 4\ 2\ 2]. \end{aligned}$$

方法二: 直接相乘.

设 $y = \ltimes_{i=1}^{m} y_i$. 已知

$$y_i = M_i x, \quad i = 1, \cdots, m,$$

这里 $x = \ltimes_{i=1}^{n} x_i$. 将所有 y_i, $i = 1, \cdots, m$ 相乘得

$$\begin{aligned} y &= M_1 x M_2 x \cdots M_m x \\ &= M_1 \left(I_{2^n} \otimes M_2\right) \left(I_{2^{2n}} \otimes M_3\right) \cdots \left(I_{2^{(m-1)n}} \otimes M_m\right) x^m \\ &= \prod_{i=1}^{m} \left(I_{2^{(i-1)n}} \otimes M_{i-1}\right) \left(M_{r,2^n}\right)^{m-1} x, \end{aligned}$$

于是

$$M_F = \prod_{i=1}^{m} \left(I_{2^{(i-1)n}} \otimes M_{i-1}\right) \left(M_{r,2^n}\right)^{m-1}. \tag{3.2.4}$$

(3.2.4) 可以作为公式来应用, 但实际上应用这个公式一般并不方便. 而直接相乘本身有时却是方便的. 见下例.

例 3.2.2　考虑下列映射 $F : (x_1, \cdots, x_n) \mapsto (y_1, \cdots, y_n)$:

$$\begin{cases} y_1 = x_1 \wedge x_2, \\ y_2 = x_2 \wedge x_3, \\ \qquad \vdots \\ y_n = x_n \wedge x_1. \end{cases} \tag{3.2.5}$$

求 F 的结构矩阵. 各式相乘得

$$\begin{aligned} y &= M_c x_1 x_2 M_c x_2 x_3 \cdots M_c x_n x_1 \\ &= M_c (I_4 \otimes M_c) \cdots (I_{4^{n-1}} \otimes M_c) x_1 x_2^2 x_3^2 \cdots x_n^2 x_1 \\ &= \prod_{i=1}^{n} \left(I_{4^{i-1}} \otimes M_c\right) x_1 W_{[2,4^{n-1}]} x_1 x_2^2 \cdots x_n^2 \\ &= \prod_{i=1}^{n} \left(I_{4^{i-1}} \otimes M_c\right) \left(I_2 \otimes W_{[2,4^{n-1}]}\right) x_1^2 x_2^2 \cdots x_n^2 \\ &= \prod_{i=1}^{n} \left(I_{4^{i-1}} \otimes M_c\right) \left(I_2 \otimes W_{[2,4^{n-1}]}\right) M_r x_1 M_r x_2 \cdots M_r x_n \\ &= \prod_{i=1}^{n} \left(I_{4^{i-1}} \otimes M_c\right) \left(I_2 \otimes W_{[2,4^{n-1}]}\right) \prod_{i=1}^{n} \left(I_{2^{i-1}} \otimes M_r\right) x. \end{aligned}$$

所以

$$M_F = \prod_{i=1}^{n} (I_{4^{i-1}} \otimes M_c) \left(I_2 \otimes W_{[2,4^{n-1}]}\right) \prod_{i=1}^{n} (I_{2^{i-1}} \otimes M_r).$$

3.3 作为逻辑算子的布尔函数

在经典逻辑中, 一个命题 (陈述), 如果它只可能是 “真” 或 “假”, 则称其为一个逻辑命题. 例如: “地球是球形的” 或 “地球是平的” 都是逻辑命题. 一个取值只能是 “真” 或 “假” 的变量称为逻辑变量. 其取值记作 $\{T, F\}$ 或 $\{1, 0\}$ ($T \sim 1$, $F \sim 0$). 因此, 所有的逻辑算子都是布尔函数. 但因为常用的逻辑算子有它们特定的逻辑意义, 有必要单独研究它们的性质. 从一般布尔函数观点看, 它们无非是一类布尔函数的性质.

除了第 2 章讨论过的一元逻辑算子: 非 ($\neg$), 二元逻辑算子: 析取 (或) ($\vee$); 合取 (与) ($\wedge$); 蕴涵 ($\rightarrow$); 等价 ($\leftrightarrow$), 之外, 以下 3 个二元逻辑算子也常用[122].

(i) 异或, 记作 $x \bar{\vee} y$, 定义为 $x \bar{\vee} y = \neg(x \leftrightarrow y)$, 它指 x 或 y 为真, 但不能同时为真.

(ii) 与非, 记作 $x \uparrow y$, 定义为 $x \uparrow y = \neg(x \wedge y)$.

(iii) 或非, 记作 $x \downarrow y$, 定义为 $x \downarrow y = \neg(x \vee y)$.

它们的结构矩阵分别为

$$\begin{aligned} M_{\bar{\vee}} &= \delta_2[2\ 1\ 1\ 2], \\ M_{\uparrow} &= \delta_2[2\ 1\ 1\ 1], \\ M_{\downarrow} &= \delta_2[2\ 2\ 2\ 1]. \end{aligned} \tag{3.3.1}$$

下面给出一些重要的逻辑等价关系.

命题 3.3.1 *存在以下逻辑等价关系:*

$$\neg\neg x = x; \tag{3.3.2}$$

$$(x \wedge y) \wedge z = x \wedge (y \wedge z); \tag{3.3.3}$$

$$(x \vee y) \vee z = z \vee (y \vee z); \tag{3.3.4}$$

$$\neg(x \wedge y) = \neg x \vee \neg y; \tag{3.3.5}$$

$$\neg(x \vee y) = \neg x \wedge \neg y; \tag{3.3.6}$$

$$x \rightarrow y = \neg x \vee y; \tag{3.3.7}$$

$$\neg(x \rightarrow y) = x \wedge \neg y; \tag{3.3.8}$$

$$x \rightarrow y = \neg y \rightarrow \neg x; \tag{3.3.9}$$

$$x \rightarrow (y \rightarrow z) = (x \wedge y) \rightarrow z; \tag{3.3.10}$$

$$\neg(x \leftrightarrow y) = x \leftrightarrow \neg y. \tag{3.3.11}$$

证明 我们举一个等式证明, 其他等式同理可证. 证 (3.3.9). 由于逻辑函数结构矩阵的唯一性, 只要证明左、右两式结构矩阵相等即可.

$$\begin{aligned}右边 &= M_i M_n y M_n x = M_i M_n (I_2 \otimes M_n) y x \\ &= M_i M_n (I_2 \otimes M_n) W_{[2]} x y.\end{aligned}$$

因为

$$\begin{aligned}&M_i M_n (I_2 \otimes M_n) W_{[2]} \\ &= \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} = M_i,\end{aligned}$$

即得 (3.3.9). □

注 命题 3.3.1 中的等式 (3.3.2)–(3.3.11) 称绝对逻辑等价, 因为对变量取连续值 $x, y, z \in [0,1]$ 时, 这些等式仍然成立[60].

命题 3.3.2 存在以下逻辑等价关系:

$$x \vee x = x; \tag{3.3.12}$$

$$x \wedge x = x; \tag{3.3.13}$$

$$y \vee (x \wedge \neg x) = y; \tag{3.3.14}$$

$$y \wedge (x \vee \neg x) = y; \tag{3.3.15}$$

$$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z); \tag{3.3.16}$$

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z); \tag{3.3.17}$$

$$x \leftrightarrow y = (x \rightarrow y) \wedge (y \rightarrow x); \tag{3.3.18}$$

$$x \leftrightarrow y \Longrightarrow (x \wedge y) \vee (\neg x \wedge \neg y). \tag{3.3.19}$$

证明 只证 (3.3.14). 设 $y = [\delta, 1-\delta]^{\mathrm{T}}$, $x = [\mu, 1-\mu]^{\mathrm{T}}$. 那么,

$$\begin{aligned}左边 &= M_\vee y M_\wedge x M_\neg x \\ &= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta \\ 1-\delta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu \\ 1-\mu \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \mu \\ 1-\mu \end{bmatrix}\end{aligned}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta \\ 1-\delta \end{bmatrix} \begin{bmatrix} \mu(1-\mu) \\ \mu^2+\mu(1-\mu)+(1-\mu)^2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta[\mu(1-\mu)] \\ \delta[\mu^2+\mu(1-\mu)+(1-\mu)^2] \\ (1-\delta)[\mu(1-\mu)] \\ (1-\delta)[\mu^2+\mu(1-\mu)+(1-\mu)^2] \end{bmatrix}$$

$$= \begin{bmatrix} \delta+(1-\delta)\mu(1-\mu) \\ (1-\delta)[\mu^2+\mu(1-\mu)+(1-\mu)^2] \end{bmatrix}.$$

因为 $\mu \in \mathcal{D}$, 我们有, 左边 $= [\delta, 1-\delta]^{\mathrm{T}} = y$. □

下面的关系式很重要, 证明不难, 留给读者.

命题 3.3.3(De Morgan 定律)

1. $$\neg(x \wedge y) = (\neg x) \vee (\neg y). \tag{3.3.20}$$
2. $$\neg(x \vee y) = (\neg x) \wedge (\neg y). \tag{3.3.21}$$

注 De Morgan 定律可推广到多个或可数个变量的情况, 即

1. $$\neg\left(\wedge_{i=1}^{\infty} x_i\right) = \vee_{i=1}^{\infty}(\neg x_i). \tag{3.3.22}$$
2. $$\neg\left(\vee_{i=1}^{\infty} x_i\right) = \wedge_{i=1}^{\infty}(\neg x_i). \tag{3.3.23}$$

定义 3.3.1 1) 一个逻辑函数称为恒真式, 如果对自变量的任何取值, 函数值恒为真.

2) 一个逻辑函数称为矛盾式, 如果对自变量的任何取值, 函数值恒为假.

3) 给定两个逻辑表达式 L_1, L_2. 如果 $L_1 \to L_2$ 为恒真式, 则称 L_1 恒真蕴涵 L_2, 记作 $L_1 \Rightarrow L_2$.

命题 3.3.4 $L_1 \Rightarrow L_2$, 当且仅当, 如果 $L_2 = \delta_2^2$ 则 $L_1 = \delta_2^2$.

证明 (充分性) 设 $L_2 = \delta_2^1$, 并记 $L_1 = \begin{bmatrix} \alpha \\ 1-\alpha \end{bmatrix}$. 那么

$$L_1 \to L_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ 1-\alpha \end{bmatrix} \delta_2^1 = \delta_2^1.$$

设 $L_2 = \delta_2^2$. 由假定 $L_1 = \delta_2^2$. 因此

$$L_1 \to L_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \delta_2^2 \delta_2^2 = \delta_2^1.$$

(必要性) 设 $L_2=\delta_2^2$ 而 $L_1=\delta_2^1$. 那么

$$L_1 \to L_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \delta_2^1 \delta_2^2 = \delta_2^2,$$

这说明 $L_1 \to L_2$ 不恒真. □

下面给出一些重要的恒真蕴涵式.

命题 3.3.5 *存在以下恒真蕴涵关系:*

$$x \wedge y \Rightarrow x; \tag{3.3.24}$$

$$x \wedge y \Rightarrow y; \tag{3.3.25}$$

$$x \Rightarrow x \vee y; \tag{3.3.26}$$

$$y \Rightarrow x \vee y; \tag{3.3.27}$$

$$\neg x \Rightarrow x \to y; \tag{3.3.28}$$

$$y \Rightarrow x \to y; \tag{3.3.29}$$

$$\neg(x \to y) \Rightarrow x; \tag{3.3.30}$$

$$\neg(x \to y) \Rightarrow \neg y; \tag{3.3.31}$$

$$\neg x \wedge (x \vee y) \Rightarrow y; \tag{3.3.32}$$

$$x \wedge (x \to y) \Rightarrow y; \tag{3.3.33}$$

$$\neg y \wedge (x \to y) \Rightarrow \neg x; \tag{3.3.34}$$

$$(x \to y) \wedge (y \to z) \Rightarrow x \to z; \tag{3.3.35}$$

$$(x \vee y) \wedge (x \to z) \wedge (y \to z) \Rightarrow z. \tag{3.3.36}$$

证明　只证 (3.3.36). 利用命题 3.3.4 并设右边为 “假”, 即, $z=\delta_2^2$, 检验左式, 有

$$\begin{aligned}
&(x \vee y) \wedge (x \to z) \wedge (y \to z) \\
&= M_\wedge M_\vee xy M_\wedge M_\to xz M_\to yz \\
&= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ 1-p \end{bmatrix} \begin{bmatrix} q \\ 1-q \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\
&\quad \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ 1-p \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} q \\ 1-q \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\end{aligned}$$

$$= \begin{bmatrix} (p+q-pq)(1-p)(1-q) \\ (p+q-pq)^2+(1-p)^2(1-q)^2+(p+q-pq)(1-p)(1-q) \end{bmatrix}.$$

我们必须检验四种情况: (i) $p=0$, $q=0$, (ii) $p=0$, $q=1$, (iii) $p=1$, $q=0$, (iv) $p=1$, $q=1$. 不难验证, 无论哪种情况, 左式均为 δ_2^2. □

最后, 给几个关于 "异或" "或非" 及 "与非" 的逻辑等式.

命题 3.3.6 下列等式为逻辑等式.

$$x\bar{\vee}y = y\bar{\vee}x; \tag{3.3.37}$$

$$(x\bar{\vee}y)\bar{\vee}z = x\bar{\vee}(y\bar{\vee}z); \tag{3.3.38}$$

$$x \wedge (y\bar{\vee}z) = (x \wedge y)\bar{\vee}(x \wedge z); \tag{3.3.39}$$

$$x\bar{\vee}y = (x \wedge \neg y) \vee (\neg x \wedge y); \tag{3.3.40}$$

$$x\bar{\vee}y = \neg(x \leftrightarrow y); \tag{3.3.41}$$

$$x \uparrow y = y \uparrow x; \tag{3.3.42}$$

$$x \downarrow y = y \downarrow x; \tag{3.3.43}$$

$$x \uparrow (y \uparrow z) = \neg x \vee (y \wedge z); \tag{3.3.44}$$

$$(x \uparrow y) \uparrow z = (x \wedge y) \vee \neg z; \tag{3.3.45}$$

$$x \downarrow (y \downarrow z) = \neg x \wedge (y \vee z); \tag{3.3.46}$$

$$(x \downarrow y) \downarrow z = (x \vee y) \wedge \neg z. \tag{3.3.47}$$

3.4 标 准 型

定义 3.4.1 设 $f(x_1,\cdots,x_n)$ 为一关于 $x_1,\cdots,x_n$ 的逻辑函数, 记

$$X = \{x_1, \neg x_1, x_2, \neg x_2, \cdots, x_n, \neg x_n\}.$$

(i) 如果 $f(x)$ 能表示成

$$\begin{aligned} f(x) = &\vee_{i=1}^{s}(\xi_1^i \wedge \xi_2^i \wedge \cdots \wedge \xi_{n_i}^i), \\ &\xi_j^i \in X,\ i=1,\cdots,s;\ j=1,\cdots,n_i. \end{aligned} \tag{3.4.1}$$

则称 (3.4.1) 为 $f(x)$ 的析取标准型.

(ii) 如果 $f(x)$ 能表示成

$$\begin{aligned} f(x) = &\wedge_{i=1}^{s}(\xi_1^i \vee \xi_2^i \vee \cdots \vee \xi_{n_i}^i), \\ &\xi_j^i \in X,\ i=1,\cdots,s;\ j=1,\cdots,n_i. \end{aligned} \tag{3.4.2}$$

则称 (3.4.2) 为 $f(x)$ 的合取标准型.

定理 3.4.1 任何一个逻辑函数都有它的析取标准型及合取标准型.

证明 设 $y = f(x_1, \cdots, x_n)$ 的结构矩阵为 $M_f \in \mathcal{L}_{2\times 2^n}$. 则有代数表达式

$$y = M_f x_1 x_2 \cdots x_n.$$

将 M_f 平分为两块:

$$M_f = \left[M_f^1 \ M_f^2\right].$$

那么, 当 $x_1 = \delta_2^1$ 时,

$$y = M_f^1 x_2 \cdots x_n;$$

当 $x_1 = \delta_2^2$ 时,

$$y = M_f^2 x_2 \cdots x_n;$$

因此, y 可以写成

$$y = (x_1 \wedge f_1(x_2, \cdots, x_n)) \vee (\neg x_1 \wedge f_2(x_2, \cdots, x_n)),$$

这里 f_i 是以 M_f^i 为结构矩阵的逻辑函数, $i = 1, 2$. 用同样的方法对 f_1 及 f_2 进行分解. 继续这个过程, 则可得到析取标准式.

至于合取标准式, 可由下法得到: 先对 $\neg f(x_1, \cdots, x_n)$ 作析取标准式. 然后取"非", 再利用 De Morgan 定律即得合取标准式. □

下面给一个例子.

例 3.4.1 考虑

$$f(x, y, z) = x \leftrightarrow (y \uparrow z). \tag{3.4.3}$$

(i) (析取标准式) 不难算出

$$M_f = \delta_2[2\ 1\ 1\ 1\ 1\ 2\ 2\ 2].$$

于是

$$f(x, y, z) = (x \wedge f_1(y, z)) \vee (\neg x \wedge f_2(y, z)),$$

这里

$$M_{f_1} = \delta_2[2\ 1\ 1\ 1]; \quad M_{f_2} = \delta_2[1\ 2\ 2\ 2].$$

显见

$$f_1(y, z) = \neg(y \wedge z) = \neg y \vee \neg z.$$

于是

$$x \wedge f_1(y,z) = [x \wedge (\neg y)] \vee [x \wedge (\neg z)]$$

类此

$$f_2(y,z) = y \wedge z.$$

于是

$$\neg x \wedge f_2(y,z) = (\neg x) \wedge y \wedge z.$$

最后得析取标准式为

$$f(x,y,z) = [x \wedge (\neg y)] \vee [x \wedge (\neg z)] \vee [(\neg x) \wedge y \wedge z].$$

(ii) (合取标准式) 对 M_f 取对偶 (即 $\delta_2^1 \to \delta_2^2,\ \ \delta_2^2 \to \delta_2^1$), 可得

$$M_{\neg f} = \delta_2[1\ 2\ 2\ 2\ 2\ 1\ 1\ 1].$$

利用前面用过的方法, 可以得到 $\neg f$ 的析取标准式为

$$\neg f(x,y,z) = (x \wedge y \wedge z) \vee [(\neg x) \wedge (\neg y)] \vee [(\neg x) \wedge (\neg z)].$$

两边取 "非", 再利用 De Morgan 定律可得 f 的合取标准式为

$$f(x,y,z) = [(\neg x) \vee (\neg y) \vee (\neg z)] \wedge (x \vee y) \wedge (x \vee z).$$

定义 3.4.2 *一组逻辑算子称为完备集, 如果任何逻辑函数都可以用它们来表示.*

由定理 3.4.1 可知 $\{\vee, \wedge, \neg\}$ 是一组完备集. 但由 De Morgan 定律

$$x \wedge y = \neg\left[(\neg x) \vee (\neg y)\right],$$

于是 $\{\vee, \neg\}$ 是一组完备集. 同样

$$x \vee y = \neg\left[(\neg x) \wedge (\neg y)\right],$$

于是 $\{\wedge, \neg\}$ 也是一组完备集.

3.5 注释与参考

布尔函数最初来自数理逻辑, 数理逻辑是整个近代数学的一个基础, 本书实际上涉及得很少, 有兴趣的读者可参考一些标准教科书, 例如文献 [84].

随后, 它在密码学、计算机科学、线路设计等许多学科中起着重要作用. 通过将逻辑变量用向量形式表示, 我们得到逻辑函数的代数表达式. 然后, 利用矩阵半张量积方法对其进行各种运算, 以代替直接的逻辑运算, 这是一个极其有效的方法, 也是本书讨论的基本原理和出发点.

第 4 章　布尔多项式

4.1　伽罗瓦域 $\mathbb{Z}_p$

定义 4.1.1　一个集合 S, 在它上面定义一个运算 $*: S \times S \to S$. $(S, *)$ 称为一个群, 如果它满足

1) (结合律)

$$a * (b * c) = (a * b) * c.$$

2) (单位元) 存在单位元 $e \in S$, 使得

$$a * e = e * a = a, \quad a \in S.$$

3) (逆元) 对每一个 $a \in S$, 存在唯一的 $a^{-1} \in S$, 使得

$$a * a^{-1} = a^{-1} * a = e.$$

一个群称为阿贝尔群, 如果它满足

4) (交换律)

$$a * b = b * a, \quad a, b \in S.$$

下面给几个简单例子.

例 4.1.1　1) 复数集 $\mathbb{C}$、实数集 $\mathbb{R}$、有理数集 $\mathbb{N}$、整数集 $\mathbb{Z}$ 对于普通数的加法, 均构成阿贝尔群. 其单位元是 0. 对 $a + bi \in \mathbb{C}$, 其逆元为 $-a - bi$. 其余类此.

2) $\mathbb{C}\backslash\{0\}$、实数集 $\mathbb{R}\backslash\{0\}$、有理数集 $\mathbb{N}\backslash\{0\}$ 对于普通数的乘法, 均构成阿贝尔群. 其单位元是 1. 对 $0 \neq a + bi \in \mathbb{C}$, 其逆元为 $\dfrac{a - bi}{a^2 + b^2}$. 其余类此. $\mathbb{Z}\backslash\{0\}$ 对于普通数的乘法, 不构成一个群, 因为当 $a \neq \pm 1$ 时其逆不存在.

3) 记 $GL(n, \mathbb{R})$ 为 $n \times n$ 可逆实矩阵集合. 它对普通矩阵乘法构成一个群. 单位元为单位阵 I_n, $A \in GL(n, \mathbb{R})$ 的逆元为 A^{-1}. 这个群称为一般线性群.

定义 4.1.2　一个集合 S, 在它上面定义两个运算: 加法 (记作 $+$), 乘法 (记作 $\times$). 如果

1) $(S, +)$ 是一个阿贝尔群, 记其单位元为 0.

2) $(S\backslash\{0\}, \times)$ 是一个阿贝尔群, 记其单位元为 1.

3) 加乘满足分配律:

$$(a + b) \times c = a \times c + b \times c,$$

则称 $(S, +, \times)$ 为一个域.

例 4.1.2 复数集 $\mathbb{C}$、实数集 $\mathbb{R}$、有理数集 $\mathbb{N}$ 对于普通数的加法及普通数的乘法均构成域, 分别称为复数域、实数域及有理数域.

设 p 为一个素数, 记

$$\mathbb{Z}_p = \{0, 1, \cdots, p-1\}.$$

在 $\mathbb{Z}_p$ 上定义加法 (记作 $\langle + \rangle$) 及乘法 (记作 $\langle \times \rangle$) 如下:

$$\begin{cases} a \langle + \rangle b := a + b \pmod p, \\ a \langle \times \rangle b := ab \pmod p. \end{cases} \tag{4.1.1}$$

那么, 不难检验 $(\mathbb{Z}_p, \langle + \rangle, \langle \times \rangle)$ 是一个域, 称其为伽罗瓦域.

例 4.1.3 考虑 $\mathbb{Z}_5$. 对加法有 (表 4.1.1)

表 4.1.1 $\mathbb{Z}_5$ 加法表

$\langle + \rangle$	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

于是有加法逆 (表 4.1.2)

表 4.1.2 $\mathbb{Z}_5$ 加法逆

x	0	1	2	3	4
$-x$	0	4	3	2	1

对乘法有 (表 4.1.3)

表 4.1.3 $\mathbb{Z}_5$ 乘法表

$\langle \times \rangle$	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

于是有乘法逆 (表 4.1.4)

表 4.1.4 $\mathbb{Z}_5$ 乘法逆

x	0	1	2	3	4
x^{-1}	—	1	3	2	4

当 $p=2$ 时, 有 $\mathbb{Z}_2=\mathcal{D}$, 称其为布尔域. 显然在 $\mathbb{Z}_2$ 中 $\langle+\rangle$ 和 $\langle\times\rangle$ 均为逻辑算子, 实际上有

$$\langle+\rangle=\bar{\vee},\quad \langle\times\rangle=\wedge.$$

一个重要的问题是: $\{\langle+\rangle,\langle\times\rangle\}$ 是否为一个完备集? 即, 是否可以用它们表示所有的布尔函数. 答案是肯定的. 首先

$$\neg x=1\,\langle+\rangle\,x,$$

而且 $\langle\times\rangle=\wedge$, 而已知 $\{\neg,\wedge\}$ 是一个完备集, 结论显见.

为了记号的简洁, 在本章也仅限本章我们约定

$$\begin{cases}a\,\langle+\rangle\,b=a+b,\\ a\,\langle\times\rangle\,b=ab,\quad a,b\in\mathbb{Z}_2.\end{cases}$$

4.2 布尔向量的表达

给定一个 n 维布尔向量, $\boldsymbol{x}=(x_1,\cdots,x_n)\in\mathbb{Z}_2^n$. 我们可以用以下三种不同方式来表示它:

(i) 向量形式:

$$V_x=(x_1,\cdots,x_n),\quad x_i\in\mathbb{Z}_2,\ i=1,\cdots,n. \tag{4.2.1}$$

(ii) 数量形式: 将 $x=\{x_1,x_2,\cdots,x_n\}$ 等同一个二进制数 $x_1x_2\cdots x_n$. 转化为十进制数, 则有

$$\chi_x=x_12^{n-1}+x_22^{n-2}+\cdots+x_n, \tag{4.2.2}$$

这里 $0\leqslant\chi_x\leqslant 2^n-1$.

(iii) 半张量形式: 利用等价表达 $1\sim\delta_2^1$ 及 $0\sim\delta_2^2$, 则 $x_i\in\Delta$. 定义 x 为其乘积:

$$x:=\ltimes_{i=1}^n x_i\in\Delta_{2^n}. \tag{4.2.3}$$

三种表达各有其方便处. 它们显然是等价的. 下面给出它们的转化关系, 证明留给读者.

命题 4.2.1 1) (从数量形式到半张量形式) 设 χ_x 已知, 则

$$x=\delta_{2^n}^{2^n-\chi_x}. \tag{4.2.4}$$

2) (从半张量形式到数量形式) 设 $x=\delta_{2^n}^t$ 已知, 则

$$\chi_x=2^n-t. \tag{4.2.5}$$

至于数量形式与向量形式间的转换, 它本质上是十进制数与二进制数的转换, 利用 (4.2.2) 即可. 下面给一个例子.

例 4.2.1 假定 $n=8$.

(i) 设 $\chi_x=51$. 那么

$$V_x=[0,0,1,1,0,0,1,1];\quad x=\delta_{2^8}^{2^8-51}=\delta_{2^8}^{205}.$$

(ii) 设 x 的向量形式 $V_x=[1,1,0,0,1,0,1,0]$. 那么

$$\chi_x=2^7+2^6+2^3+2^1=202;\quad x=\delta_{2^8}^{2^8-202}=\delta_{2^8}^{54}.$$

(iii) 设 x 的半张量形式为 $x=\delta_{2^8}^{120}$. 那么

$$\chi_x=2^8-120=136;\quad V_x=[1,0,0,0,1,0,0,0].$$

下面考虑布尔函数. 设布尔函数 $f:\mathbb{Z}_2^n\to\mathbb{Z}_2$. 由前两章的讨论可知, 存在唯一的矩阵 $M_f\in\mathcal{L}_{2\times 2^n}$, 称为 f 的结构矩阵, 使在自变量的半张量形式下 f 可表示为

$$y:=f(x_1,\cdots,x_n)=M_f\ltimes_{i=1}^n x_i,\quad x_i\in\varDelta. \tag{4.2.6}$$

当自变量 $(x_1,\cdots,x_n)$ 表示为数量形式时, 函数可以表示为 (对自变量所有可能值的) 向量形式

$$V_f:=(f(0),f(1),\cdots,f(2^n-1)). \tag{4.2.7}$$

这两种表达形式有如下关系:

命题 4.2.2 记 M_f 的第一行为 m^f, 即, $m^f=\mathrm{Row}_1(M_f)$. 那么

$$m_i^f=(V_f)_{2^n+1-i},\quad i=1,\cdots,2^n. \tag{4.2.8}$$

注意, 实质上它们只是前后顺序颠倒一下而已.

例 4.2.2 1) 设 $M_f=\delta_2[1\ 1\ 2\ 1\ 2\ 1\ 1\ 2]$. 那么

$$V_f=(0\ 1\ 1\ 0\ 1\ 0\ 1\ 1).$$

2) 设 $V_f=(1\ 1\ 0\ 1\ 1\ 0\ 1\ 0)$. 那么

$$M_f=\delta_2[2\ 1\ 2\ 1\ 1\ 2\ 1\ 1].$$

最后, 再回忆一下 $x\in\mathbb{Z}_2^n$ 的三种等价表示:

$$\begin{aligned}&V_x=(x_1,\cdots,x_n)^{\mathrm{T}},\quad x_i\in\mathcal{D}\\&\Leftrightarrow x=\ltimes_{i=1}^n x_i=\delta_{2^n}^{2^n-\chi_x},\quad x_i\in\varDelta\\&\Leftrightarrow \chi_x=x_1 2^{n-1}+x_2 2^{n-2}+\cdots+x_n=2^n-t,\ \text{当}\ x=\delta_{2^n}^t.\end{aligned}$$

实际上, 在密码学的许多文献中, 它们都用同一符号 x 表示. 本章中, 我们用 $\boldsymbol{x}$ 表示 V_x, 用 $\boldsymbol{f}$ 表示 V_f. 而 χ_x 和半张量形式都用 x 表示.

4.3 布尔多项式

所谓布尔多项式即用 $\mathbb{Z}_2$ 上的多项式形式来表示布尔函数, 它在应用上有许多方便之处. 为此先引入一些记号.

定义 4.3.1　1) 设 $\boldsymbol{x}$, $\boldsymbol{c} \in \mathbb{Z}_2^n$ 为两向量, $\boldsymbol{x} = (x_1, \cdots, x_n)$ 及 $\boldsymbol{c} = (c_1, \cdots, c_n)$. 作为记号, 定义

$$x_i^1 := x_i, \quad x_i^0 = \neg x_i. \tag{4.3.1}$$

于是有

$$x_i^{c_i} = \begin{cases} 1, & x_i = c_i, \\ 0, & x_i \neq c_i. \end{cases} \tag{4.3.2}$$

2) 定义

$$\boldsymbol{x}^{\boldsymbol{c}} := \prod_{i=1}^{n} x_i^{c_i} = \begin{cases} 1, & \boldsymbol{x} = \boldsymbol{c} \\ 0, & \boldsymbol{x} \neq \boldsymbol{c}. \end{cases} \tag{4.3.3}$$

严格地说, (4.3.1) 和 (4.3.3) 的第一式为定义, (4.3.2) 和 (4.3.3) 的第二个等式是由定义导出的结果. 利用这些记号可得

命题 4.3.1　考虑布尔函数 $f: \mathbb{Z}_2^n \to \mathbb{Z}_2$. 变量 $\boldsymbol{x} = (x_1, \cdots, x_n) \in \mathbb{Z}_2^n$. 则有如下表达式:

1) (幂形式)

$$f(x) = \sum_{i=0}^{2^n-1} f(i)\boldsymbol{x}^{\boldsymbol{i}}. \tag{4.3.4}$$

这里 $\boldsymbol{i}$ 是将 i 从 $\chi_x = i$ 转化为向量形式 V_x.

2) (多项式形式)

$$\begin{aligned} f(x) &= a_0 + a_1x_1 + \cdots a_nx_n + a_{12}x_1x_2 \\ &\quad + \cdots + a_{n-1\,n}x_{n-1}x_n + \cdots + a_{12\cdots n}x_1x_2\cdots x_n \\ &= a_0 + \sum_{k=1}^{n} \sum_{1 \leqslant j_1 < \cdots < j_k \leqslant n} a_{j_1\cdots j_k}x_{j_1}\cdots x_{j_k}. \end{aligned} \tag{4.3.5}$$

证明　注意到 $\boldsymbol{x}^{\boldsymbol{x}} = 1$, 以及 $\boldsymbol{x}^{\boldsymbol{i}} = 0$, $\boldsymbol{i} \neq \boldsymbol{x}$. 于是, 由 (4.3.3) 可直接推出 (4.3.4) .

根据定义有

$$x_i^0 = \begin{cases} 1, & x_i = 0, \\ 0, & x_i = 1. \end{cases}$$

因此,

$$x_i^0 = x_i + 1.$$

于是 (4.3.5) 可由 (4.3.4) 通过将 x_i^0 替换为 $x_i + 1$, 再将所有因子相乘而得到. □

(4.3.5) 称为 $f(x)$ 的布尔多项式形式. $f(x)$ 的次数, 记作 $\deg(f(x))$, 指的是 $f(x)$ 的布尔多项式形式的次数. 当 $\deg(f(x)) = 1$ 时, $f(x)$ 称为 x 的仿射函数. 一个仿射函数, 如果常数项 $a_0 = 0$, 则称为 x 的线性函数.

在某些文献中, 仿射函数被统称为线性函数.

例 4.3.1 考虑

$$f(x_1, x_2, x_3) = (x_1 \wedge x_2) \leftrightarrow x_3.$$

容易算得

$$\begin{array}{lll} f(0) = f(0,0,0) = 1, & f(1) = f(0,0,1) = 0, & f(2) = f(0,1,0) = 1, \\ f(3) = f(0,1,1) = 0, & f(4) = f(1,0,0) = 1, & f(5) = f(1,0,1) = 0, \\ f(6) = f(1,1,0) = 0, & f(7) = f(1,1,1) = 1. & \end{array}$$

于是, f 的向量形式为

$$\boldsymbol{f} = [1\ 0\ 1\ 0\ 1\ 0\ 0\ 1].$$

f 的布尔多项式形式为

$$\begin{aligned} f(x) &= x_1^0 x_2^0 x_3^0 + x_1^0 x_2^1 x_3^0 + x_1^1 x_2^0 x_3^0 + x_1^1 x_2^1 x_3^1 \\ &= (1 + x_1)(1 + x_2)(1 + x_3) + (1 + x_1) x_2 (1 + x_3) \\ &\quad + x_1 (1 + x_2)(1 + x_3) + x_1 x_2 x_3 \\ &= 1 + x_3 + x_1 x_2. \end{aligned}$$

下面引入几个记号:

1) $\mathcal{B}_n^{\mathcal{F}}$: $\mathbb{Z}_2^n \to \mathbb{Z}_2$ 的布尔函数集, 它是 $\mathbb{Z}_2$ 上的一个向量空间.

2) $\mathcal{B}_n^{\mathcal{A}}$: $\mathcal{B}_n^{\mathcal{F}}$ 的仿射子空间.

3) $\mathcal{B}_n^{\mathcal{L}}$: $\mathcal{B}_n^{\mathcal{A}}$ 的线性子空间.

下面讨论布尔函数 f 的布尔多项式形式与结构矩阵形式间的转换. f 可表示如下:

$$
\begin{aligned}
f(x) &= M_f x \\
&= m_f \begin{bmatrix} x_1^1 \\ x_1^0 \end{bmatrix} \begin{bmatrix} x_2^1 \\ x_2^0 \end{bmatrix} \cdots \begin{bmatrix} x_n^1 \\ x_n^0 \end{bmatrix} \\
&= m_f \begin{bmatrix} x_1 \\ x_1+1 \end{bmatrix} \begin{bmatrix} x_2 \\ x_2+1 \end{bmatrix} \cdots \begin{bmatrix} x_n \\ x_n+1 \end{bmatrix} \\
&= m_f \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \end{bmatrix} \cdots \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ x_n \end{bmatrix} \\
&= m_f \left(\underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}}_{n} \right) \begin{bmatrix} 1 \\ x_1 \end{bmatrix} \begin{bmatrix} 1 \\ x_2 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ x_n \end{bmatrix} \\
&:= m_f P_n \xi_n := \alpha \xi_n,
\end{aligned} \tag{4.3.6}
$$

这里

$$
\alpha = m_f P_n, \tag{4.3.7}
$$

$$
P_n = \left(\underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}}_{n} \right), \tag{4.3.8}
$$

$$
\xi_n = \begin{bmatrix} 1 \\ x_1 \end{bmatrix} \begin{bmatrix} 1 \\ x_2 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ x_n \end{bmatrix} \tag{4.3.9}
$$

是 $\mathbb{Z}_2^n$ 上的多项式的一组基. 这样, (4.3.6)–(4.3.9) 就将 $f(x)$ 从半张量积形式 (M_f) 变到多项式形式 ($\alpha\xi_n$).

当然, 上述多项式形式 ($\alpha\xi_n$) 还是有点不自然, 人们或许希望将它写成普通的升幂多项式的样子, 即

$$
f(x) = \beta \eta_n, \tag{4.3.10}
$$

这里, η_n 是按字典序和升幂形式排列的基底, 即

$$\eta_n = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \\ x_1x_2 \\ \vdots \\ x_{n-1}x_n \\ x_1x_2x_3 \\ \vdots \\ x_{n-2}x_{n-1}x_n \\ \vdots \\ x_1x_2\cdots x_n \end{bmatrix}. \tag{4.3.11}$$

我们需要找出 η_n 与 ξ_n 的关系. 为了这个目的, 首先要知道 $x_{i_1}x_{i_2}\cdots x_{i_n}$ 在 ξ_n 中的位置. 记 $\mu_{i_1,i_2,\cdots,i_r}$, $i_1 < i_2 < \cdots < i_r$ 为 $x_{i_1}x_{i_2}\cdots x_{i_r}$ 在 ξ_n 中的位置, 考察 ξ_n 中的因子乘积 (从后往前乘).

情形 1 (两个因子乘积):

$$\begin{bmatrix} 1 \\ x_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ x_n \end{bmatrix} = \begin{bmatrix} 1 \\ x_n \\ x_{n-1} \\ x_{n-1}x_n \end{bmatrix}. \tag{4.3.12}$$

于是有 $\mu_n = 2^0 + 1$, $\mu_{n-1} = 2^1 + 1$, $\mu_{n-1,n} = 2^1 + 2^0 + 1$. 由半张量积定义, 这四项的位置再与前面因子相乘时是不动的.

情形 2 (三个因子乘积):

$$\begin{bmatrix} 1 \\ x_{n-2} \end{bmatrix} \begin{bmatrix} 1 \\ x_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ x_n \end{bmatrix} = \begin{bmatrix} 1 \\ x_{n-2} \end{bmatrix} \begin{bmatrix} 1 \\ x_n \\ x_{n-1} \\ x_{n-1}x_n \end{bmatrix} = \begin{bmatrix} 1 \\ x_n \\ x_{n-1} \\ x_{n-1}x_n \\ x_{n-2} \\ x_{n-2}x_n \\ x_{n-2}x_{n-1} \\ x_{n-2}x_{n-1}x_n \end{bmatrix}. \tag{4.3.13}$$

这八项的位置再与前面因子乘时是不动的.

⋮

情形 2^n (全部乘积):

$$\begin{bmatrix}1\\x_1\end{bmatrix}\begin{bmatrix}1\\x_2\end{bmatrix}\cdots\begin{bmatrix}1\\x_n\end{bmatrix}=\xi_n. \tag{4.3.14}$$

观察以上乘积, 可得

定理 4.3.1

$$\mu_{i_1,i_2,\cdots,i_r}=\sum_{j=1}^{r}2^{n-i_j}+1. \tag{4.3.15}$$

证明　对任一 j, x_{n-j} 在逆序乘积中首次出现的位置是 $\mu_{n-j}=2^j+1$.

然后, 对任意项 $x_{n-i_1}x_{n-i_2}\cdots x_{n-i_r}$, 可依次找到序列元素$x_{n-i_1}, x_{n-i_1}x_{n-i_2},\cdots$, $x_{n-i_1}x_{n-i_2}\cdots x_{n-i_r}$ 的位置, 从而不难验证公式 (4.3.15). □

根据定理 4.3.1, 构造 Φ_n 矩阵如下:

$$\Phi_n=\delta_{2^n}[1,\phi_1,\phi_2,\cdots,\phi_n], \tag{4.3.16}$$

其中

$$\phi_r=(\mu_{1,2,\cdots,r},\mu_{2,3,\cdots,r+1},\cdots,\mu_{n-r+1,n-r+2,\cdots,n}),\quad r=1,2,\cdots,n,$$

这里, $n_r=\binom{n}{r}=\dfrac{n!}{r!(n-r)!}$.

容易证明

$$\Phi_n^{\mathrm{T}}\xi_n=\eta_n. \tag{4.3.17}$$

最后, 注意到 $f=\alpha\xi_n=\beta\eta_n$, 于是有

$$\begin{cases}\alpha=m_fP_n,\\ \beta=\alpha\Phi_n=m_fP_n\Phi_n.\end{cases} \tag{4.3.18}$$

4.4　Walsh 变换

定义 4.4.1　设 $\boldsymbol{x}=(x_1,\cdots,x_n)$, $\boldsymbol{\omega}=(\omega_1,\cdots,\omega_n)\in\mathbb{Z}_2^n$.

1) $\boldsymbol{x}$ 和 $\boldsymbol{\omega}$ 的内积定义为

$$\boldsymbol{x}\cdot\boldsymbol{\omega}=x_1\omega_1+\cdots+x_n\omega_n\in\mathbb{Z}_2. \tag{4.4.1}$$

2) 给定 $\boldsymbol{\omega} \in \mathbb{Z}_2^n$. 定义一个函数 $Q_{\boldsymbol{\omega}} : \mathbb{Z}_2^n \to \{-1, 1\}$ 如下:

$$Q_{\boldsymbol{\omega}}(\boldsymbol{x}) = (-1)^{\boldsymbol{\omega} \cdot \boldsymbol{x}}, \tag{4.4.2}$$

其向量形式为

$$\boldsymbol{Q_{\omega}} = (Q_{\boldsymbol{\omega}}(0), Q_{\boldsymbol{\omega}}(1), \cdots, Q_{\boldsymbol{\omega}}(2^n - 1)).$$

引理 4.4.1 设 $\omega \neq 0$. 那么

$$\sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega} \cdot \boldsymbol{x}} = 0. \tag{4.4.3}$$

证明 设 $\omega_i \neq 0$. 对每一个 $\boldsymbol{x} = (x_1, \cdots, x_n)$ 满足 $\boldsymbol{\omega} \cdot \boldsymbol{x} = 0$, 我们构造一个 $\boldsymbol{x}^* = (x_1, \cdots, \neg x_i, \cdots, x_n)$, 它满足 $\boldsymbol{\omega} \cdot \boldsymbol{x}^* = 1$. 由于 $\boldsymbol{x} \leftrightarrow \boldsymbol{x}^*$ 是一个一对一的映射, 于是可知

$$\big|\{\boldsymbol{x} \,|\, \boldsymbol{\omega} \cdot \boldsymbol{x} = 0\}\big| = \big|\{\boldsymbol{x} \,|\, \boldsymbol{\omega} \cdot \boldsymbol{x} = 1\}\big|.$$

这就证明了 (4.4.3). □

命题 4.4.1 $\{\boldsymbol{Q_{\omega}} \,|\, \omega = 0, 1, \cdots, 2^n - 1\}$ 为一族正交函数. 准确地说,

$$\boldsymbol{Q_{\alpha}} \cdot \boldsymbol{Q_{\beta}} = \begin{cases} 2^n, & \boldsymbol{\alpha} = \boldsymbol{\beta}, \\ 0, & \boldsymbol{\alpha} \neq \boldsymbol{\beta}. \end{cases} \tag{4.4.4}$$

证明 当 $\boldsymbol{\alpha} = \boldsymbol{\beta}$ 时. 则

$$\begin{aligned} \boldsymbol{Q_{\alpha}} \cdot \boldsymbol{Q_{\beta}} &= \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\alpha} \cdot \boldsymbol{x}} (-1)^{\boldsymbol{\beta} \cdot \boldsymbol{x}} \\ &= \sum_{x=0}^{2^n-1} (-1)^{2\boldsymbol{\alpha} \cdot \boldsymbol{x}} \\ &= \sum_{x=0}^{2^n-1} (-1)^0 = 2^n. \end{aligned}$$

当 $\boldsymbol{\alpha} \neq \boldsymbol{\beta}$ 时. 由于 $\boldsymbol{\alpha} \neq \boldsymbol{\beta}$, 则 $\boldsymbol{\alpha} + \boldsymbol{\beta} \neq \boldsymbol{0}$. 利用引理 4.4.1 可得

$$\boldsymbol{Q_{\alpha}} \cdot \boldsymbol{Q_{\beta}} = \sum_{x=0}^{2^n-1} (-1)^{(\boldsymbol{\alpha} + \boldsymbol{\beta}) \cdot \boldsymbol{x}} = 0.$$

□

由于 $\{\boldsymbol{Q}_\omega \,|\, \omega = 0, 1, \cdots, 2^n - 1\}$ 是一组正交基, 那么, 对任一布尔函数 $f \in \mathcal{B}_n^{\mathcal{F}}$, 其向量形式可表示为

$$\boldsymbol{f} = \sum_{\omega=0}^{2^n-1} S_f(\omega)\boldsymbol{Q}_\omega. \tag{4.4.5}$$

定义 4.4.2　$\{S_f(\omega) \,|\, \omega = 0, 1, \cdots, 2^n - 1\}$ 称为 f 的第一 Walsh 变换.

命题 4.4.2　第一 Walsh 变换可计算如下:

$$S_f(\omega) = 2^{-n} \sum_{x=0}^{2^n-1} f(x) Q_x(\omega). \tag{4.4.6}$$

证明　对给定的 $\omega_0 \in \mathbb{Z}_2^n$, 有

$$\begin{aligned}
\boldsymbol{f} \cdot \boldsymbol{Q}_{\omega_0} &= \left(\sum_{\omega=0}^{2^n-1} S_f(\omega)\boldsymbol{Q}_\omega(x) \right) \cdot \boldsymbol{Q}_{\omega_0}(x) \\
&= S_f(\omega_0)\boldsymbol{Q}_{\omega_0}(x) \cdot \boldsymbol{Q}_{\omega_0}(x) \\
&= 2^n S_f(\omega_0).
\end{aligned} \tag{4.4.7}$$

另一方面, 有

$$\boldsymbol{f} \cdot \boldsymbol{Q}_{\omega_0} = \sum_{x=0}^{2^n-1} Q_{\omega_0}(x) f(x).$$

因此,

$$S_f(\omega_0) = 2^{-n} \sum_{x=0}^{2^n-1} Q_{\omega_0}(x) f(x).$$

□

下面考虑另一种 Walsh 变换. 定义

$$g(x) := 1 - 2f(x). \tag{4.4.8}$$

考察 $\boldsymbol{g}$ 对基底 $\{\boldsymbol{Q}_\omega | \omega = 0, 1, \cdots, 2^n - 1\}$ 的展开:

$$\boldsymbol{g} = \sum_{\omega=0}^{2^n-1} S_{(f)}(\omega)\boldsymbol{Q}_\omega(x). \tag{4.4.9}$$

同样, 对给定的 $\boldsymbol{\omega}_0 \in \mathbb{Z}_2^n$, 有

$$\boldsymbol{g} \cdot \boldsymbol{Q}_{\omega_0} = \sum_{x=0}^{2^n-1} (1 - 2f(x)) Q_{\omega_0}(x). \tag{4.4.10}$$

容易看出

$$(-1)^{f(x)} = 1 - 2f(x). \tag{4.4.11}$$

因此

$$\boldsymbol{g} \cdot \boldsymbol{Q}_{\omega_0} = \sum_{x=0}^{2^n-1} (-1)^{f(x)} Q_{\omega_0}(x). \tag{4.4.12}$$

另一方面, 不难得到

$$\boldsymbol{g} \cdot \boldsymbol{Q}_{\omega_0} = \left(\sum_{\omega=0}^{2^n-1} S_{(f)}(\omega) Q_\omega(x) \right) \cdot Q_{\omega_0}(x) = 2^n \times S_{(f)}(\omega_0).$$

于是

$$S_{(f)}(\omega) = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} Q_\omega(x).$$

定义 4.4.3 给定一个布尔函数 $f(x)$,

$$S_{(f)}(\omega) = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} Q_\omega(x), \quad \omega = 0, 1, \cdots, 2^n - 1, \tag{4.4.13}$$

称为 f 的第二 Walsh 变换.

利用 (4.4.8) 及 (4.4.9) 可得

$$f(x) = \frac{1}{2} - \frac{1}{2} \sum_{\omega=0}^{2^n-1} S_{(f)}(\omega) Q_\omega(x). \tag{4.4.14}$$

下面讨论两种 Walsh 变换的关系. 有

命题 4.4.3 $S_{(f)}(\omega)$ 和 $S_f(\omega)$ 具有下面的关系

$$S_{(f)}(\omega) = \begin{cases} -2S_f(\omega), & \boldsymbol{\omega} \neq \boldsymbol{0}, \\ 1 - 2S_f(\omega), & \boldsymbol{\omega} = \boldsymbol{0}. \end{cases} \tag{4.4.15}$$

证明 利用 (4.4.11), 有

$$\begin{aligned} S_{(f)}(\omega) &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} Q_\omega(x) \\ &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} (1 - 2f(x)) Q_\omega(x) \\ &= \frac{1}{2^n} \sum_{x=0}^{2^n-1} Q_\omega(x) - \frac{2}{2^n} \sum_{x=0}^{2^n-1} f(x) Q_\omega(x). \end{aligned} \tag{4.4.16}$$

根据引理 4.4.1 可推出 (4.4.15). □

下面讨论 Walsh 变换的一些性质.

命题 4.4.4 设 $S_f(\omega)$ 为函数 $f(x)$ 的 Walsh 变换, 那么对任何 $a \in \mathbb{Z}_2^n$, $f(x+a)$ 的 Walsh 变换为

$$S_{f(x+a)}(\omega) = Q_a(\omega) S_f(\omega). \tag{4.4.17}$$

证明 由定义, $f(x+a)$ 的 Walsh 变换为

$$\begin{aligned}
S_{f(x+a)}(\omega) &= 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} f(x+a) \\
&= (-1)^{2\boldsymbol{\omega}\cdot\boldsymbol{a}} \cdot 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} f(x+a) \\
&= (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{a}} \cdot 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot(\boldsymbol{x}+\boldsymbol{a})} f(x+a) \\
&= (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{a}} \cdot 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} f(x) \\
&= Q_a(\omega) S_f(\omega).
\end{aligned}$$

□

命题 4.4.5 设 $S_f(\omega)$ 为 $f(x)$ 的 Walsh 变换, $S_g(\omega)$ 为 $g(x)$ 的 Walsh 变换. 那么 $af(x) + bg(x)$ 的 Walsh 变换为

$$S_{af+bg}(\omega) = aS_f(\omega) + bS_g(\omega). \tag{4.4.18}$$

证明 从定义出发, $af(x) + bg(x)$ 的 Walsh 变换为

$$\begin{aligned}
S_{af+bg}(\omega) &= 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} (af(x) + bg(x)) \\
&= a \cdot 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} f(x) + b \cdot 2^{-n} \sum_{x=0}^{2^n-1} (-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}} g(x) \\
&= aS_f(\omega) + bS_g(\omega).
\end{aligned}$$

□

命题 4.4.6 (Plancheral 公式)

$$\sum_{\omega=0}^{2^n-1} S_f^2(\omega) = S_f(0). \tag{4.4.19}$$

证明 由于

$$S_f(\omega) = 2^{-n}\sum_{x=0}^{2^n-1} Q_\omega(x)f(x),$$

可推得

$$\begin{aligned}\sum_{\omega=0}^{2^n-1} S_f^2(\omega) &= 2^{-2n}\sum_{\omega=0}^{2^n-1}\sum_{x=0}^{2^n-1} Q_\omega(x)f(x)S_f(\omega)\\ &= 2^{-2n}\sum_{x=0}^{2^n-1} f(x)\left(\sum_{\omega=0}^{2^n-1} Q_\omega(x)S_f(\omega)\right)\\ &= 2^{-2n}\sum_{x=0}^{2^n-1} f^2(x)\\ &= 2^{-2n}\sum_{x=0}^{2^n-1} f(x).\end{aligned}$$

另一方面,

$$S_f(0) = 2^{-2n}\sum_{\omega=0}^{2^n-1} f(x)Q_x(0) = 2^{-2n}\sum_{\omega=0}^{2^n-1} f(x).$$

因此

$$\sum_{\omega=0}^{2^n-1} S_f^2(\omega) = S_f(0).$$

□

由两种 Walsh 变换的关系可得

命题 4.4.7(Parseval 公式) 对第二 Walsh 变换有

$$\sum_{\omega=0}^{2^n-1} S_{(f)}^2(\omega) = 1. \tag{4.4.20}$$

下面考虑 f 和它的 Walsh 变换 S_f 之间转换的矩阵表达式. 根据对称性, 记

$$Q(\omega, x) := Q_\omega(x).$$

根据 (4.4.5), 有

$$\begin{aligned}&(f(0), f(1), \cdots, f(2^n-1))\\ =\ &(S_f(0), S_f(1), \cdots, S_f(2^n-1))\end{aligned}$$

$$\times \begin{bmatrix} Q(0,0) & Q(0,1) & \cdots & Q(0,2^n-1) \\ Q(1,0) & Q(1,1) & \cdots & Q(1,2^n-1) \\ & & & \\ Q(2^n-1,0) & Q(2^n-1,1) & \cdots & Q(2^n-1,2^n-1) \end{bmatrix}$$

$$:= (S_f(0), S_f(1), \cdots, S_f(2^n-1))H_n. \tag{4.4.21}$$

上式可以简记成

$$\boldsymbol{f} = \boldsymbol{S}_f H_n. \tag{4.4.22}$$

定义 4.4.4 给定 $A = (a_{ij}) \in \mathcal{M}_{s\times s}$. A 称为 Hadamard 矩阵, 如果它满足

(i)

$$a_{ij} = \pm 1, \quad 1 \leqslant i, j \leqslant s;$$

(ii)

$$A^{\mathrm{T}}A = AA^{\mathrm{T}} = sI_s.$$

命题 4.4.8 在 (4.4.21) 中定义的转换矩阵 H_n 满足

(i)

$$H_{n+1} = H_1 \otimes H_n; \tag{4.4.23}$$

(ii)

$$H_n H_n = 2^n I_{2^n}; \tag{4.4.24}$$

(iii) H_n 是一个 Hadamard 矩阵.

证明 (ii) 是命题 4.4.1 的直接推论. 因此 (iii) 是显见的. 只需证明 (i). 考虑 H_{n+1}. 它可以表示为

$$H_{n+1} = \begin{bmatrix} H_n^1 & H_n^2 \\ H_n^2 & H_n^3 \end{bmatrix},$$

这里

$$H_n^1 = \begin{bmatrix} Q(0,0) & \cdots & Q(0,2^n-1) \\ \vdots & & \vdots \\ Q(2^n-1,0) & \cdots & Q(2^n-1,2^n-1) \end{bmatrix};$$

$$H_n^2 = \begin{bmatrix} Q(0,2^n) & \cdots & Q(0,2^{n+1}-1) \\ \vdots & & \vdots \\ Q(2^n-1,2^n) & \cdots & Q(2^n-1,2^{n+1}-1) \end{bmatrix};$$

$$H_n^3 = \begin{bmatrix} Q(2^n, 2^n) & \cdots & Q(2^n, 2^{n+1}-1) \\ \vdots & & \vdots \\ Q(2^{n+1}-1, 2^n) & \cdots & Q(2^{n+1}-1, 2^{n+1}-1) \end{bmatrix}.$$

考察 H_n^1, 记其为

$$H_n^1 = \left(Q^1(\omega, x)\right),$$

其中

$$\boldsymbol{x} = (0, x_1, x_2, \cdots, x_{2^n}), \quad \boldsymbol{\omega} = (0, \omega_1, \omega_2, \cdots, \omega_{2^n}),$$

并且让 $(x_1, x_2, \cdots, x_{2^n})$ 和 $(\omega_1, \omega_2, \cdots, \omega_{2^n})$ 取遍从 $(0, \cdots, 0)$ 到 $(1, \cdots, 1)$ 的值. 那么, 容易看出 $H_n^1 = H_n$.

其次, 考察 H_n^2, 记其为

$$H_n^2 = \left(Q^2(\omega, x)\right),$$

其中

$$\boldsymbol{x} = (1, x_1, x_2, \cdots, x_{2^n}), \quad \boldsymbol{\omega} = (0, \omega_1, \omega_2, \cdots, \omega_{2^n}),$$

同样, 让 $(x_1, x_2, \cdots, x_{2^n})$ 和 $(\omega_1, \omega_2, \cdots, \omega_{2^n})$ 取遍从 $(0, \cdots, 0)$ 到 $(1, \cdots, 1)$ 的值. 那么, 可得 $H_n^2 = H_n$.

最后考察 H_n^3, 记其为

$$H_n^3 = \left(Q^3(\omega, x)\right),$$

其中

$$\boldsymbol{x} = (1, x_1, x_2, \cdots, x_{2^n}), \quad \boldsymbol{\omega} = (1, \omega_1, \omega_2, \cdots, \omega_{2^n}).$$

类似前面的讨论可知 $H_n^3 = -H_n$.

注意到

$$H(1) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

于是, 不难看出

$$H_{n+1} = H_1 \otimes H_n.$$

□

4.5 线 性 结 构

定义 4.5.1 设 $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ 为一布尔函数.

1) $a \in \mathbb{Z}_2^n$ 称为 f 的不变线性结构, 如果

$$f(x+a)+f(x)=0. \tag{4.5.1}$$

2) $a \in \mathbb{Z}_2^n$ 称为 f 的变线性结构, 如果

$$f(x+a)+f(x)=1. \tag{4.5.2}$$

3) 定义

$$\begin{aligned} E_0 &:= \{a \in \mathbb{Z}_2^n \mid f(x+a)+f(a)=0\}, \\ E_1 &:= \{a \in \mathbb{Z}_2^n \mid f(x+a)+f(a)=1\}, \\ E &:= E_0 \cup E_1. \end{aligned} \tag{4.5.3}$$

那么 E 称为 f 的线性结构子空间.

命题 4.5.1　1) $E_0 \cap E_1 = \varnothing$.

2) E 是一个向量空间, 而 $E_0 \subset E$ 是其线性子空间.

3) 如果 $E_1 \neq \varnothing$. 那么 $E_1 = E_0 + a$, 这里 $a \in E_1$ 为任一元素.

证明　1) 直接来自定义.

2) •　设 $a, b \in E_0$. 那么

$$\begin{aligned} & f(x+a+b)+f(x) \\ =& f(x+a+b)+f(x+a)+f(x+a)+f(x) \\ =& 0+0=0. \end{aligned}$$

即, $a+b \in E_0 \subset E$.

•　令 $a \in E_0$, $b \in E_1$. 那么

$$\begin{aligned} & f(x+a+b)+f(x) \\ =& f(x+a+b)+f(x+a)+f(x+a)+f(x) \\ =& 1+0=1. \end{aligned}$$

即, $a+b \in E_1 \subset E$.

•　令 $a, b \in E_1$. 那么

$$\begin{aligned} & f(x+a+b)+f(x) \\ =& f(x+a+b)+f(x+a)+f(x+a)+f(x) \\ =& 1+1=0. \end{aligned}$$

即, $a+b \in E_0 \subset E$.

因此 E 是一个向量空间. 而且, 情况 1 说明 E_0 是其线性子空间.

3) 前面证明了: 如果 $a \in E_0$, $b \in E_1$, 那么 $a+b \in E_1$. 即, $E_0 + b \subset E_1$. 现在假定 $\xi \in E_1$, 那么 $\xi + b \in E_0$, 于是

$$\xi = (\xi + b) + b \in E_0 + b,$$

即 $E_1 \subset E_0 + b$. 立得结论. □

我们有如下推论.

推论 4.5.1 1)

$$|E_0| = 2^r, \tag{4.5.4}$$

这里, r 是 E_0 的维数.

2) 或者 $E_1 = \varnothing$, 或者

$$|E_1| = |E_0|. \tag{4.5.5}$$

证明 1) 作为向量空间, (4.5.4) 显见.

2) 设 $E_1 \neq \varnothing$, 且令 $b \in E_1$. 定义映射 $\pi_b : E_0 \to E_1$ 如下: $x \mapsto x + b$, 容易验证 π 是一一映上的. □

定义 4.5.2 给定一个布尔函数 f.

(i) 设 $|E| = 2^q$. 那么 q 称为 f 的线性结构的维数. 如果 $q > 0$, 则称 f 为具有线性结构的布尔函数.

(ii) 对具有线性结构的布尔函数 f, 如果 $E_0 \neq \{0\}$, 则称 f 为 I 型, 如果 $E_0 = \{0\}$, 称 f 为 II 型.

下面讨论如何计算 E_0 和 E_1. 设布尔函数 $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ 的结构矩阵为 $M_f \in \mathcal{L}_{2\times 2^n}$. 记 $\alpha = \ltimes_{i=1}^n a_i$, $x = \ltimes_{i=1}^n x_i$. 那么, 不难发现 $(a_1, \cdots, a_n) \in E_1$, 当且仅当

$$M_f M_p a_1 x_1 M_p a_2 x_2 \cdots M_p a_n x_n = M_f x_1 x_2 \cdots x_n, \tag{4.5.6}$$

这里, $M_p = \delta_2[2\ 1\ 1\ 2]$ 为 "+" 的结构矩阵. 而且, 直接计算可验证 (4.5.6) 等价于

$$M_f M_p \ltimes_{i=1}^{n-1} (I_{2^{2i}} \otimes M_p) \ltimes_{i=1}^{n-1} \left(I_{2^i} \otimes W_{[2,2^i]}\right) \alpha x = M_f x. \tag{4.5.7}$$

定义

$$\Psi_f := M_f M_p \ltimes_{i=1}^{n-1} (I_{2^{2i}} \otimes M_p) \ltimes_{i=1}^{n-1} \left(I_{2^i} \otimes W_{[2,2^i]}\right),$$

并且, 将它等分成 2^n 块如下:

$$\Psi_f = [\psi_1\ \psi_2\ \cdots\ \psi_{2^n}],$$

这里 $\psi_k = \mathrm{Blk}_k(\Psi_f)$, $k = 1, 2, \cdots, 2^n$. 那么, 直接计算即可得到如下结论.

定理 4.5.1 令 $\alpha = \ltimes_{j=1}^{n} a_j := \delta_{2^n}^{i}$. 那么

1) $(a_1, \cdots, a_n) \in E_0$, 当且仅当 $\psi_i = M_f$.

2) $(a_1, \cdots, a_n) \in E_1$, 当且仅当 $\psi_i = M_n M_f$.

例 4.5.1 1) 设 f 的结构矩阵为

$$M_f = \delta_2[2\ 2\ 2\ 1\ 2\ 1\ 1\ 1].$$

那么

$$M_n M_f = \delta_2[1\ 1\ 1\ 2\ 1\ 2\ 2\ 2].$$

容易计算

$$\begin{aligned}\Psi_f = [&1\ \ 1\ \ 1\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 1\ \ 1\ \ 2\ \ 1\ \ 2\ \ 1\ \ 2\ \ 2\\ &1\ \ 2\ \ 1\ \ 1\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 1\ \ 1\ \ 1\ \ 2\ \ 2\ \ 2\ \ 1\\ &1\ \ 2\ \ 2\ \ 2\ \ 1\ \ 1\ \ 1\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 1\ \ 1\ \ 2\ \ 1\\ &2\ \ 2\ \ 1\ \ 2\ \ 1\ \ 2\ \ 1\ \ 1\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 1\ \ 1\ \ 1].\end{aligned}$$

由 $\psi_8 = M_f$, 而 $\psi_1 = M_n M_f$, 我们知道

$$[0, 0, 0, 0, 0, 0, 0, 0] \in E_0,$$

以及

$$[1, 1, 1, 1, 1, 1, 1, 1] \in E_1.$$

由 $|E_0| = |E_1| = 1$, 我们知道 $|E| = 2$, 即 $\dim(E) = 1$. 所以 f 是 I 型具有线性结构的布尔函数.

2) 假定 f 的结构矩阵为

$$M_f = \delta_2[2\ 2\ 2\ 2\ 2\ 2\ 2\ 1].$$

那么

$$M_n M_f = \delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 2].$$

容易计算

$$\begin{aligned}\Psi_f = [&1\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\\ &2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 2\\ &2\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\\ &2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 1\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 2\ \ 1].\end{aligned}$$

只有 $\psi_8 = M_f$, 有

$$E_0 = \{(0, 0, 0, 0, 0, 0, 0, 0)\},$$

且

$$E_1 = \varnothing.$$

现在有 $|E_0| = 1$ 及 $|E_1| = 0$, 因此, $|E| = 1$, $\dim(E) = 0$. f 也是 I 型具有线性结构的布尔函数.

4.6 非线性性

记 $\mathcal{B}_n^{\mathcal{F}}$ 为 n 元布尔函数集合; $\mathcal{B}_n^{\mathcal{A}}$ 为 n 元布尔仿射函数集合.

定义 4.6.1 1) 设 $\boldsymbol{x} = (x_1, \cdots, x_n) \in \mathbb{Z}_2^n$. 那么, $\boldsymbol{x}$ 的汉明权重 (Hamming weight) 定义为

$$w_H(\boldsymbol{x}) = \big|\{i \,|\, x_i \neq 0\}\big|. \tag{4.6.1}$$

2) 给定布尔函数 $f, g \in \mathcal{B}_n^{\mathcal{F}}$. 那么, f 和 g 的汉明距离定义为

$$d_H(f, g) := w_H(\boldsymbol{f} + \boldsymbol{g}). \tag{4.6.2}$$

下面考虑布尔函数的非线性性.

定义 4.6.2 设 $f \in \mathcal{B}_n^{\mathcal{F}}$.

1) f 的非线性性, 记作 N_f, 定义为

$$N_f := \min_{\ell \in \mathcal{B}_n^{\mathcal{A}}} d_H(\boldsymbol{f}, \boldsymbol{\ell}). \tag{4.6.3}$$

2) f 的线性性, 记作 C_f, 定义为

$$C_f := \max_{\ell \in \mathcal{B}_n^{\mathcal{A}}} d_H(\boldsymbol{f}, \boldsymbol{\ell}). \tag{4.6.4}$$

定义 4.6.3 设 $\ell(x) \in \mathcal{B}_n^{\mathcal{A}}$ 满足

$$d_H(\boldsymbol{\ell}, \boldsymbol{f}) = N_f. \tag{4.6.5}$$

那么, $\ell(x)$ 称为 f 的最佳线性逼近.

给定 $f \in \mathcal{B}_n^{\mathcal{F}}$, 为计算其非线性性, 我们考虑 f 的线性等价. 下面的定理通过 Walsh 变换给出布尔函数线性等价的概率[11].

定理 4.6.1 设 $\boldsymbol{\omega} = (\omega_1, \omega_2 \cdots \omega_n)$, $\boldsymbol{x} = (x_1, x_2 \cdots x_n) \in \mathbb{Z}_2^n$, 且 $x \in \mathbb{Z}_2^n$ 均匀分布. 那么

$$P\left(\{x | f(x) = \boldsymbol{\omega} \cdot \boldsymbol{x}\}\right) = \frac{1 + S_{(f)}(\omega)}{2}, \tag{4.6.6}$$

$$P\left(\{x | f(x) \neq \boldsymbol{\omega} \cdot \boldsymbol{x}\}\right) = \frac{1 - S_{(f)}(\omega)}{2}. \tag{4.6.7}$$

证明　因为

$$\begin{aligned}S_{(f)}(x) &= 2^{-1}\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{\boldsymbol{\omega}\cdot\boldsymbol{x}}\\ &= 2^{-1}\left(\sum_{\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}}(-1)^{f(x)+\boldsymbol{\omega}\cdot\boldsymbol{x}}+\sum_{\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}}(-1)^{f(x)+\boldsymbol{\omega}\cdot\boldsymbol{x}}\right)\\ &= 2^{-1}\left(\sum_{\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}}1+\sum_{\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}}(-1)\right)\\ &= P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right)-P\left(\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right),\end{aligned}$$

且 $P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right)+P\left(\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right)=1$, 即可得到 (4.6.6) 及 (4.6.7). □

定理 4.6.2　设 $f\in\mathcal{B}_n^{\mathcal{F}}$, 且记

$$a=\max_{0\leqslant\omega\leqslant 2^n-1}\left|S_{(f)}(\omega)\right|.$$

那么

$$N_f=2^n\left(\frac{1-a}{2}\right); \tag{4.6.8}$$

且

$$C_f=2^n\left(\frac{1+a}{2}\right). \tag{4.6.9}$$

证明　对任何仿射函数 $\ell(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}+\omega_0$ 有

$$\begin{aligned}&P\left(\{x|f(x)=\ell(x)\}\right)\\ =&P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}+\omega_0\}\right)\\ =&\begin{cases}P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right), & \omega_0=0,\\ P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}+1\}\right)=P\left(\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right), & \omega_0=1.\end{cases}\end{aligned}$$

因此

$$P\left(\{x|f(x)=\ell(x)\}\right)=\begin{cases}P\left(\{x|f(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right), & \ell(x)=\boldsymbol{\omega}\cdot\boldsymbol{x},\\ P\left(\{x|f(x)\neq\boldsymbol{\omega}\cdot\boldsymbol{x}\}\right), & \ell(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}+1.\end{cases}$$

根据定理 4.6.1, 有

$$P\left(\{x|f(x)=\ell(x)\}\right)=\begin{cases}\dfrac{1+S_{(f)}(\omega)}{2}, & \ell(x)=\boldsymbol{\omega}\cdot\boldsymbol{x},\\ \dfrac{1-S_{(f)}(\omega)}{2}, & \ell(x)=\boldsymbol{\omega}\cdot\boldsymbol{x}+1.\end{cases}$$

于是可得

$$\max_{\ell \in \mathcal{B}_n^{\mathcal{A}}} P\left(\{x|f(x)=\ell(x)\}\right)=\frac{1+a}{2}; \tag{4.6.10}$$

以及

$$\min_{\ell \in \mathcal{B}_n^{\mathcal{A}}} P\left(\{x|f(x)=\ell(x)\}\right)=\frac{1-a}{2}. \tag{4.6.11}$$

利用 (4.6.10) 可得

$$\begin{aligned} N_f &= \min_{\ell \in \mathcal{B}_n^{\mathcal{A}}} w_H(\boldsymbol{f}+\boldsymbol{\ell}) \\ &= \min_{\ell \in \mathcal{B}_n^{\mathcal{A}}} 2^n P\left(\{x|f(x)\neq \ell(x)\}\right) \\ &= \min_{\ell \in \mathcal{B}_n^{\mathcal{A}}} 2^n \left(1-P\left(\{x|f(x)=\ell(x)\}\right)\right) \\ &= 2^n\left(1-\max_{\ell \in \mathcal{B}_n^{\mathcal{A}}} P\left(\{x|f(x)=\ell(x)\}\right)\right) \\ &= 2^n\left(\frac{1-a}{2}\right). \end{aligned}$$

利用 (4.6.10), 类似前面的讨论可得

$$C_f=2^n\left(\frac{1+a}{2}\right).$$

□

一个显然的推论是

推论 4.6.1 对任一 $f \in \mathcal{B}_n^{\mathcal{F}}$,

$$N_f+C_f=2^n. \tag{4.6.12}$$

由定理 4.6.2 不难看出, 当 a 为最小时其相应的 N_f 为最大. 因为

$$\sum_{\omega=0}^{2^n-1} S_{(f)}^2=1,$$

当 $\left|S_{(f)}\right|=\text{const.}$ 时, a 达到最小. 这时有

$$\left|S_{(f)}\right|=2^{-\frac{n}{2}}. \tag{4.6.13}$$

因此可知

$$N_f=2^n\left(\frac{1-2^{-\frac{n}{2}}}{2}\right)=2^{n-1}-2^{\frac{n}{2}-1}. \tag{4.6.14}$$

综上可得下面的结论.

命题 4.6.1

$$N_f \leqslant 2^{n-1} - 2^{\frac{n}{2}-1}. \tag{4.6.15}$$

当 (4.6.14) 成立时, f 具有最高的非线性度. 这样的布尔函数称弯曲函数，也称完全非线性函数. 它们在密码学中有许多应用[37].

4.7 布尔函数的对称性

记 $\boldsymbol{S}_n$ 为 n 阶对称群, $\boldsymbol{H}_n < \boldsymbol{S}_n$ 为 $\boldsymbol{S}_n$ 的子群.

定义 4.7.1 一个布尔函数 $f(x) \in \mathcal{B}_n^{\mathcal{F}}$ 称为关于 $\boldsymbol{H}_n$ 对称, 如果

$$f(x_{\sigma(1)}, \cdots, x_{\sigma(n)}) = f(x_1, \cdots, x_n), \quad \forall \sigma \in \boldsymbol{H}_n. \tag{4.7.1}$$

记 $\pi_i(f)$ 为 $f(x)$ 的 i 次齐次分量.

定理 4.7.1 f 关于 $\boldsymbol{H}_n$ 对称, 当且仅当 $\pi_i(f), i = 0, 1, \cdots, n$ 关于 $\boldsymbol{H}_n$ 对称.

证明 充分性是显然的, 我们证明必要性. 设 f 关于 $\boldsymbol{H}_n$ 对称. 我们用归谬法证明. 假设存在一个 i, 使 $\pi_i(f)$ 不是关于 $\boldsymbol{H}_n$ 对称的. 不妨假定 $i > 0$ 是最小的使 $\pi_i(f)$ 不对称的 i. 将 $\pi_i(f)$ 表示为

$$\pi_i(f) = \sum_{1 \leqslant j_1 < j_2 < \cdots < j_i \leqslant n} c_{j_1 \cdots j_i} x_{j_1} \cdots x_{j_i}. \tag{4.7.2}$$

注意: 不是所有的 i 次项系数系数 $c_{j_1 \cdots j_i}$ 均为 0. 否则 $\pi_i(f) = 0$, 因而是对称的. 设某个 $c_{j_1 \cdots j_i} = 1$, 如果对所有的 $\sigma \in \mathcal{H}$, $c_{\sigma(j_1) \cdots \sigma(j_i)} = 1$, 则无须证明. 故假定存在 $\sigma \in \boldsymbol{H}_n$ 使得 $c_{\sigma(j_1) \cdots \sigma(j_i)} = 0$. 定义 $x_0 = (x_1, \cdots, x_n)$ 如下

$$x_j = \begin{cases} 1, & j \in \{j_1, \cdots, j_i\}, \\ 0, & \text{其他}. \end{cases}$$

那么, 有

$$\begin{array}{rcl} \pi_i(f)(x_1, \cdots, x_n) & = & 1, \\ \pi_i(f)(x_{\sigma(1)}, \cdots, x_{\sigma(n)}) & = & 0. \end{array}$$

注意到

$$\pi_k(x_0) = 0, \quad k > i.$$

因此

$$f(x_1, \cdots, x_n) \neq f(x_{\sigma(1)}, \cdots, x_{\sigma(n)}).$$

矛盾. □

下面的推论是显见的.

推论 4.7.1 f 关于 $\boldsymbol{S}_n$ 对称, 当且仅当, 对每个 $1 \leqslant i \leqslant n-1$, i 次齐次项的系数都一样, 即

$$c_{j_1 \cdots j_i}, \quad 1 \leqslant j_1 < j_2 < \cdots < j_i \leqslant n$$

或全为 1, 或全为 0.

关于 $\boldsymbol{S}_n$ 对称也称全对称.

推论 4.7.2 $\mathcal{B}_n^{\mathcal{F}}$ 中共有 2^{n+1} 个全对称布尔函数.

证明 根据推论 4.7.1, 如果 f 关于 $\boldsymbol{S}_n$ 对称, 那么, 或者 $\pi_i(f) \equiv 0$ 或者 $\pi_i(f) \not\equiv 0$. 准确地说, 或者 $\pi_i(f) = \mathbf{0}_{d_i}$ 或者 $\pi_i(f) = \mathbf{1}_{d_i}$. 结论显见. □

如果我们将这 2^{n+1} 个全对称函数记作 $f_0, f_1, \cdots, f_{2^{n+1}-1}$, 这里 f_i 按以下方法定义: 将 i 转换成二进制数

$$i \sim i_n i_{n-1} \cdots i_0.$$

那么

$$f_i(x) = \sum_{j=0}^{n} i_j P_j(x), \quad i = 0, 1, \cdots, 2^{n+1} - 1, \tag{4.7.3}$$

这里

$$P_j(x) = \sum_{1 \leqslant k_1 < \cdots < k_j \leqslant n} \prod_{s=1}^{j} x_{k_s}.$$

于是我们有以下结果:

命题 4.7.1 设 $\{f_i \mid i = 0, 1, \cdots, 2^{n+1} - 1\}$ 为全对称布尔函数族, 依上述方法排序. 那么, f_i 的汉明权重为

$$w_H(\boldsymbol{f}_i) = \sum_{j=0}^{n} i_j \binom{n}{j}, \quad i = 0, 1, \cdots, 2^{n+1} - 1. \tag{4.7.4}$$

证明 因为

$$w_H(\boldsymbol{P}_j) = \binom{n}{j}, \quad j = 0, 1, \cdots, n,$$

利用公式 (4.7.3) 即得结论. □

为进一步研究关于 $\boldsymbol{H}_n$ 的对称性, 引入一些新概念.

定义 4.7.2 设 G 为一群, S 为一非空集. 一个映射 $G \times S \to S$ 称为 G 在 S 上的群作用, 如果它满足

(i)

$$e(s) = s, \quad \forall s \in S,$$

这里 e 是 G 的单位元.

(ii)

$$g_1(g_2(s)) = (g_1g_2)(s), \quad \forall s \in S.$$

定义 4.7.3 设群 G 作用在 S 上, 且 $s \in S$.

1) s 在 G 作用下的群轨道定义为

$$Gs := \{gs \,|\, g \in G\}. \tag{4.7.5}$$

2) s 的稳定子群定义为

$$G_s := \{g \in G \,|\, g(s) = s\}. \tag{4.7.6}$$

显然, 所有的轨道 $\{Gs \,|\, s \in S\}$ 构成 S 的一个分割. 因为对两个元素 $s_1, s_2 \in S$, 或者 $Gs_1 = Gs_2$ 或者 $Gs_1 \cap Gs_2 = \varnothing$.

关于群作用有如下性质[72]:

命题 4.7.2 一个元素 $s \in S$ 的群轨道 Gs 长度为

$$|Gs| = \frac{|G|}{|G_s|}. \tag{4.7.7}$$

轨道个数可由下述定理确定.

定理 4.7.2(Burnside 引理) 设 G 作用于 S, 形成 m 条轨道. 那么

$$m|G| = \sum_{g \in G} |\mathrm{Fix}(g)|. \tag{4.7.8}$$

这里 $\mathrm{Fix}(g)$ 为 g 的不动点集, 即

$$\mathrm{Fix}(g) = \{s \,|\, g(s) = s\}.$$

构造一个集合序列如下

$$S_i = \{\, \{j_1, \cdots, j_i\} \subset \mathbb{Z} \,|\, 1 \leqslant j_t \leqslant n, t = 1, \cdots, i; j_p \neq j_q, p \neq q\}, \\ i = 1, \cdots, n-1.$$

设 $\boldsymbol{H}_n < \boldsymbol{S}_n$. $\boldsymbol{H}_n$ 在 S_i 上的作用定义如下

$$\sigma(\{j_1, \cdots, j_i\}) := \{\sigma(j_1), \cdots, \sigma(j_i)\}. \tag{4.7.9}$$

利用定理 4.7.1 和推论 4.7.1 的证明方法, 不难推出以下结果.

定理 4.7.3 设 $\boldsymbol{H}_n$ 作用在 S_i 上的轨道数为 $m_i, i=1,\cdots,n-1$. 那么, 关于 $\boldsymbol{H}_n$ 对称的布尔函数的数目为

$$m = 2^{2+\sum\limits_{i=1}^{n-1} m_i}. \tag{4.7.10}$$

设 $\boldsymbol{C}_n < \boldsymbol{S}_n$ 是由 $(1,2,\cdots,n)$ 生成的子群, 即, $\boldsymbol{C}_n = \langle(1,2,\cdots,n)\rangle$, 它称为循环群. 一个布尔函数 $f \in \mathcal{B}_n^{\mathcal{F}}$ 称循环对称, 如果它关于 $\boldsymbol{C}_n$ 对称.

例 4.7.1 设 $n=4$, 且 $\boldsymbol{C}_4 = \langle(1,2,3,4)\rangle$ 为由 $(1,2,3,4)$ 生成的循环群. 那么

$$\begin{aligned}
S_1 &= \{1,2,3,4\};\\
S_2 &= \{\{1,2\},\{1,3\},\{1,4\},\{2,3\},\{2,4\},\{3,4\}\};\\
S_3 &= \{\{1,2,3\},\{1,2,4\},\{1,3,4\},\{2,3,4\}\}.
\end{aligned}$$

容易检验, S_1 和 S_3 各只有一条轨道. S_2 有两条轨道, 分别为

$$\begin{aligned}
&\{1,2\} \to \{2,3\} \to \{3,4\} \to \{4,1\} \to;\\
&\{1,3\} \to \{2,4\} \to .
\end{aligned}$$

因此, 可得出结论: 设 $f \in \mathcal{B}_4^{\mathcal{F}}$ 关于 $\boldsymbol{C}_4 = G\{(1,2,3,4)\}$ 对称, 则它可表示为

$$\begin{aligned}
f(x) = {}& a_0 + a_1(x_1+x_2+x_3+x_4) + a_2(x_1x_2+x_2x_3+x_3x_4+x_4x_1)\\
&+a_3(x_1x_3+x_2x_4)+a_4(x_1x_2x_3+x_1x_2x_4+x_1x_3x_4+x_2x_3x_4),
\end{aligned} \tag{4.7.11}$$

这里 $a_i = 0$ 或 1, $i=0,1,2,3,4$.

4.8 注释与参考

布尔多项式通常也称为布尔函数. 布尔函数从理论上说似乎更广一点, 但由于取值的有限性, 任何布尔函数均可以用多项式表示. 而真正有实用价值的是这种多项式表示形式. 它的主要用途是在密码学中, 用于编码[11, 37]. 本章的基本概念及部分结论来自文献 [11, 37], 大部分内容是基于文献 [152], 目的是用半张量积重新刻画和讨论布尔函数. 目前, 用半张量积方法研究布尔函数及编码问题是一个新兴方向, 有兴趣的读者可参见文献 [13, 78].

第5章　布尔代数

5.1　布尔代数

定义 5.1.1　一个集合 S, 带有两个二元算子 $\sqcup$, $\sqcap$, 一个一元算子 $'$, 和两个特殊元 0, $1 \in S$, $0 \neq 1$, 称为一个布尔代数, 如果它们满足

1) (交换律)

$$x \sqcup y = y \sqcup x; \tag{5.1.1}$$

$$x \sqcap y = y \sqcap x. \tag{5.1.2}$$

2) (结合律)

$$x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z; \tag{5.1.3}$$

$$x \sqcap (y \sqcap z) = (x \sqcap y) \sqcap z. \tag{5.1.4}$$

3) (分配律)

$$x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z); \tag{5.1.5}$$

$$x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z). \tag{5.1.6}$$

4) (归一律)

$$x \sqcup 0 = x; \tag{5.1.7}$$

$$x \sqcap 1 = x. \tag{5.1.8}$$

5) (补律)

$$x \sqcup x' = 1; \tag{5.1.9}$$

$$x \sqcap x' = 0. \tag{5.1.10}$$

由定义, 一个布尔代数可表示成 $(S,\ \sqcup,\ \sqcap,\ ',\ 1,\ 0)$.

在一个布尔代数里, 如果将 $\sqcup$ 与 $\sqcap$ 交换, 同时也将 1 与 0 交换, 结果还是一个布尔代数. 这个性质称为对偶原理.

下面举几个布尔代数的例子.

例 5.1.1 1) (经典逻辑) $S=\mathcal{D}=\{1,0\}$, $\sqcup=\vee$, $\sqcap=\wedge$, $'=\neg$. 容易检验, 它是一个布尔代数.

2) (集合运算) 设 W 为一非空集合, 考察它的子集族. 令 $S=\mathcal{P}(W)$ (通常用 $\mathcal{P}(W)$ 或 2^W 表示 W 的子集族). 定义 $\sqcup=\cup$, $\sqcap=\cap$, $'=c$ (A^c 表示集合 A 的余集); $1=W$, $0=\varnothing$. 那么, 集合运算构成一个布尔代数.

3) (布尔矩阵) 设 $S=\mathcal{B}_{m\times n}$ 为 $m\times n$ 的布尔矩阵集合, $A=(a_{ij})$, $B=(b_{ij})\in S$.

$$
\begin{aligned}
A\sqcup B&=(a_{ij}\vee b_{ij});\\
A\sqcap B&=(a_{ij}\wedge b_{ij});\\
A'&=(a'_{ij})
\end{aligned}
$$

1 为元素全为 1 的矩阵, 0 为元素全为 0 的矩阵. 容易验证 $S=\mathcal{B}_{m\times n}$ 为一布尔代数.

下面讨论布尔代数的一些基本性质:

定理 5.1.1 给定布尔代数. 则

1) (幂等律)

$$x\sqcup x=x; \tag{5.1.11}$$

$$x\sqcap x=x. \tag{5.1.12}$$

2) (归一律 (2))

$$x\sqcup 1=1; \tag{5.1.13}$$

$$x\sqcap 0=0. \tag{5.1.14}$$

3) (吸收律)

$$(x\sqcap y)\sqcup x=x; \tag{5.1.15}$$

$$(x\sqcup y)\sqcap x=x. \tag{5.1.16}$$

证明 只证 (5.1.10), 其余的留给读者.

$$
\begin{aligned}
x\sqcup x&=(x\sqcup x)\sqcap 1 && \text{根据 (5.1.8)}\\
&=(x\sqcup x)\sqcap(x\sqcup x') && \text{根据 (5.1.9)}\\
&=x\sqcap(x\sqcup x') && \text{根据 (5.1.6)}\\
&=x\sqcup 0 && \text{根据 (5.1.10)}\\
&=x. && \text{根据 (5.1.7)}
\end{aligned}
$$

□

引理 5.1.1　给定布尔代数. 如果 $x \sqcup y = 1$ 且 $x \sqcap y = 0$, 那么

$$y = x'.$$

证明

$$\begin{aligned}
y &= y \sqcup 0 && \text{根据 (5.1.7)}\\
&= y \sqcup (x \sqcap x') && \text{根据 (5.1.10)}\\
&= (y \sqcup x) \sqcap (y \sqcup x') && \text{根据 (5.1.5)}\\
&= (x \sqcup y) \sqcap (x' \sqcup y) && \text{根据 (5.1.1)}\\
&= 1 \sqcap (x' \sqcup y) && \text{根据假设}\\
&= (x \sqcup x') \sqcap (x' \sqcup y) && \text{根据 (5.1.9)}\\
&= (x' \sqcup x) \sqcap (x' \sqcup y) && \text{根据 (5.1.1)}\\
&= x' \sqcup (x \sqcap y) && \text{根据 (5.1.5)}\\
&= x' \sqcup 0 && \text{根据假设}\\
&= x'. && \text{根据 (5.1.7)}
\end{aligned}$$

□

推论 5.1.1　给定布尔代数. 则

$$(x')' = x, \quad \forall x. \tag{5.1.17}$$

证明　设 $y = x'$. 则 $y \sqcup x = x' \sqcup x = 1$; 且 $y \sqcap x = x' \sqcap x = 0$. 由引理 5.1.1,

$$x = y' = (x')'.$$

□

定理 5.1.2　布尔代数满足 De Morgan 定律, 即

$$(x \sqcup y)' = x' \sqcap y'; \tag{5.1.18}$$

$$(x \sqcap y)' = x' \sqcup y'. \tag{5.1.19}$$

证明　直接计算可得

$$\begin{aligned}
(x \sqcup y) \quad &\sqcup \quad (x' \sqcap y') = 1,\\
(x \sqcup y) \quad &\sqcap \quad (x' \sqcap y') = 0.
\end{aligned}$$

由引理 5.1.1 即得 (5.1.18). 利用 (5.1.18), 并分别用 x, y 代替 x', y', 则得

$$x \sqcap y = (x' \sqcup y')'.$$

两边取补, 即得 (5.1.19).　□

5.2 布尔代数的合成与分解

先考虑布尔代数的合成. 设 $B_1 := (S_1, \sqcup_1, \sqcap_1, '_1, 1_1, 0_1)$ 及 $B_2 := (S_2, \sqcup_2, \sqcap_2, '_2, 1_2, 0_2)$ 为两个布尔代数, 考察它们的笛卡儿积

$$S_1 \times S_2 = \{(s_1, s_2) | s_1 \in S_1, \ s_2 \in S_2\}.$$

定义运算 $\sqcup = \sqcup_1 \times \sqcup_2$, 即

$$(s_1, s_2) \sqcup (t_1, t_2) := (s_1 \sqcup_1 t_1, s_2 \sqcup_2 t_2), \quad (s_1, s_2), (t_1, t_2) \in S_1 \times S_2.$$

类此定义 $\sqcap = \sqcap_1 \times \sqcap_2$, $' = '_1 \times '_2$, $1 = (1_1, 1_2)$, $0 = (0_1, 0_2)$. 那么, 容易验证

$$B_1 \times B_2 := (S_1 \times S_2, \ \sqcup_1 \times \sqcup_2, \ \sqcap_1 \times \sqcap_2, \ '_1 \times '_2, \ (1_1, 1_2), \ (0_1, 0_2))$$

是一个布尔代数, 称为 B_1 和 B_2 的乘积代数.

显见, 这可以推广到多个布尔代数的情况的.

下面给布尔代数中的元素定义一个序.

定义 5.2.1 设 S 为一布尔代数的元素集. $x, y \in S$.

1) $x \leqslant y$ (或等价地称 $y \geqslant x$), 当且仅当, $x \sqcup y = y$.

2) $x < y$ (或等价地称 $y > x$), 当且仅当, $x \leqslant y$ 且 $x \neq y$.

条件 $x \sqcup y = y$ 可以换成 $x \sqcap y = x$. 这是因为, 容易证明

命题 5.2.1 在一个布尔代数中, $x \sqcup y = y$, 当且仅当 $x \sqcap y = x$.

不管是 $\leqslant$ 或 $<$, 对于一般的布尔代数都是偏序, 或者说 S 在这两个序底下是偏序集. 也就是说, 对任意两个元素 $x, y \in S$, 可能 $x \leqslant y$ 或 $y \leqslant x$ 均不成立. 偏序具有传递性:

命题 5.2.2 在一个布尔代数中,

(i) 如果 $x \leqslant y$ 且 $y \leqslant z$, 则 $x \leqslant z$.

(ii) 如果 $x \leqslant y$ 且 $y \leqslant x$, 则 $x = y$.

(iii) 如果 $x < y$ 且 $y < z$, 则 $x < z$.

证明 只证 (i):

$$\begin{aligned} z &= y \sqcup z \\ &= (x \sqcup y) \sqcup z \\ &= x \sqcup (y \sqcup z) \\ &= x \sqcup z. \end{aligned}$$

因此, $x \leqslant z$. □

下面的命题是显见的.

命题 5.2.3 在一个布尔代数中,

(i)

$$x \sqcap y \leqslant x \leqslant x \sqcup y. \tag{5.2.1}$$

(ii)

$$0 \leqslant x \leqslant 1. \tag{5.2.2}$$

定义 5.2.2 在一个布尔代数中, 一个非零元素 $a \in S$ 称为一个原子, 如果 a 不能写成 $a = b \sqcup c$, 这里 $a \neq b$ 且 $a \neq c$.

例 5.2.1 考察例 5.1.1.

1) (经典逻辑) 这里, 唯一的原子是 1.

2) (集合运算) 这里的原子是单元素集 $\{\{s\}|s \in S\}$.

3) (布尔矩阵) $\mathcal{B}_{m\times n}$ 中的原子是 $A = (a_{ij})$, 满足条件: 存在 (i_0, j_0), 使

$$a_{ij} = \begin{cases} 1, & i = i_0, \text{ 且 } j = j_0, \\ 0, & \text{其他}. \end{cases}$$

命题 5.2.4 $a \neq 0$ 是原子, 当且仅当, 不存在 x 使 $0 < x < a$.

证明 (必要性) 设 a 为原子. 令 $x < a$. 则 $x \sqcup a = a$, 且 $x \sqcap a = x$. 于是

$$a = a \sqcap 1 = (x \sqcup a) \sqcap (x \sqcup x') = x \sqcup (a \sqcap x').$$

因为 a 是原子, x 与 $(a \sqcap x')$ 中必有一个为 a. 但 $x < a$, 故 $a = a \sqcap x'$. 于是

$$x = a \sqcap x = (a \sqcap x') \sqcap x = a \sqcap (x' \sqcap x) = 0.$$

(充分性) 设 a 不是原子, 则 $a = x \sqcup y$, 这里 $x \neq a$, $y \neq a$. 于是 $0 \leqslant x \leqslant a$. 注意, 这里 $x \neq 0$, 否则 $a = x \sqcup y = 0 \sqcup y = y$, 矛盾. 又 $x \neq a$, 于是

$$0 < x < a.$$

□

推论 5.2.1 设 a 和 b 为两原子, 如果 $a \sqcap b \neq 0$, 则 $a = b$. 换言之, 如果 $a \neq b$, 则 $a \sqcap b = 0$.

证明 由于 $a \sqcap b \neq 0$, 则有 $0 < a \sqcap b \leqslant a$. 但 a 是原子, 故 $a \sqcap b = a$. 同理 $a \sqcap b = b$. 于是 $a = b$. □

下面的定理十分重要. (为方便, 我们也用 B 表示其元素集合, 即 $B = S$.)

定理 5.2.1 设 B 为一有限布尔代数, $A = \{a_1, \cdots, a_n\}$ 为它的原子集. 那么, 每一个 $0 \neq x \in B$ 都可以表示成

$$x = a_{i_1} \sqcup a_{i_2} \sqcup \cdots \sqcup a_{i_k}. \tag{5.2.3}$$

而且, 如果不计排列顺序, 则表达式 (5.2.3) 是唯一的.

证明 先证每一个非零元能写成原子并的形式 (5.2.3). 如果 x 是原子, 那它已属 (5.2.3) 形式. 记 $N \subset B$ 为 B 中不能写成 (5.2.3) 形式的非零元素, 证 $N = \varnothing$. 设 $x \in N$. 因为 x 不是原子, 类似命题 5.2.4 充分性证明可知, 存在 $0 < y < x$ 和 $0 < z < x$, 使得 $x = y \sqcup z$. 那么, y 和 z 中至少有一个属于 N, 否则 $x \notin N$. 设 $y \in N$, 定义 $x_1 = y$, 则 $x > x_1 > 0$. 对 x_1 作同样讨论, 可得 $x_1 > x_2 > 0$. 继续这个过程, 可得一序列 $x_i > 0, i = 1, 2, \cdots$, 使得

$$x > x_1 > x_2 > \cdots .$$

但 B 是有限集, 矛盾.

下面证唯一性. 设 x 有两种原子并:

$$x = a_1 \sqcup a_1 \sqcup \cdots \sqcup a_p = b_1 \sqcup b_1 \sqcup \cdots \sqcup b_q,$$

我们要证明 $P = Q$, 这里

$$P = \{a_1, a_2, \cdots, a_p\}; \quad Q = \{b_1, b_2, \cdots, b_q\}.$$

因为

$$\begin{aligned} a_1 &= a_1 \sqcap x = a_1 \sqcap (b_1 \sqcup b_1 \sqcup \cdots \sqcup b_q) \\ &= (a_1 \sqcap b_1) \sqcup (a_1 \sqcap b_2) \sqcup \cdots (a_1 \sqcap b_q), \end{aligned}$$

于是, 至少有一个 i, 使

$$a_1 \sqcap b_i \neq 0.$$

由推论 5.2.1, $a_1 = b_i$, 即 $a_1 \in Q$, 同理 $a_i \in Q$, $i = 2, \cdots, p$, 故 $P \subset Q$. 同理可证 $Q \subset P$. 故 $P = Q$. □

推论 5.2.2 设 $A = \{a_1, \cdots, a_n\}$ 为布尔代数 B 的原子集, 则

$$1 = a_1 \sqcup a_2 \sqcup \cdots \sqcup a_n.$$

证明 设 a_i 不属于 1 的原子并表达式 (5.2.3). 则 $1 \sqcup a_i \neq 1$, 矛盾. □

现在设 $A=\{a_1,\cdots,a_n\}$ 为布尔代数 B 的原子集, 那么, 每一个 $x\in B$ 可以用一个 n 维向量 $V_x=(v_1,v_2,\cdots,v_n)^{\mathrm{T}}\in\mathcal{B}_{n\times 1}$ 表示, 这里 $v_i=1$, 如果 a_i 出现在 x 的表达式 (5.2.3) 中, 否则 $v_i=0$. 因为表达式 (5.2.3) 唯一, 当 $\{a_i\}$ 的顺序确定后 V_x 唯一. 称 $a_i\in x$, 如果 a_i 出现在 x 的表达式 (5.2.3) 中.

现在定义投影

$$\pi_i(x):=\pi_i(V_x)=v_i,\quad x\in B,\quad i=1,\cdots,n.$$

那么, 如果 $a_i\in x$, $v_i=1$, 否则 $v_i=0$. 故 $B_i:=\pi_i(B)=\{1,0\}$, $i=1,\cdots,n$. 在 B_i 上定义

$$\begin{aligned}
\pi_i(x)\sqcup_i\pi_i(y)&:=\pi_i(x\sqcup y);\\
\pi_i(x)\sqcap_i\pi_i(y)&:=\pi_i(x\sqcap y);\\
\pi_i(x)^{'_i}&:=\pi_i(x');\\
1_i&=\pi_i(1);\\
0_i&=\pi_i(0).
\end{aligned}$$

可以证明, 以上的 $\sqcup_i$, $\sqcap_i$, $'_i$, 1_i, 0_i 是唯一定义好的, 它们使得 $(B_i,\sqcup_i,\sqcap_i,'_i,1_i,0_i)$ 成为一个二元布尔代数. 而且, $B=B_1\times B_2\times\cdots\times B_n$. 于是有

定理 5.2.2　任何一个有限布尔代数都是由它的原子生成的二元布尔代数的笛卡儿积.

5.3　二元布尔代数

由于任何一个有限布尔代数都是一些二元布尔代数的笛卡儿积. 二元布尔代数的结构就变得十分重要. 本节讨论二元布尔代数的结构. 记 $B=\mathcal{D}=\{1,0\}$. 用向量表示 $1\sim\delta_2^1$, $0\sim\delta_2^2$. 设

$$\begin{aligned}
M_{\sqcup}&=\delta_2[\alpha\ \beta\ \delta\ \gamma];\\
M_{\sqcap}&=\delta_2[a\ b\ d\ c];\\
M_{'}&=\delta_2[p,q].
\end{aligned}$$

它们须满足定义 5.1.1 中的条件.

由交换律

$$M_{\sqcup}xy=M_{\sqcup}yx=M_{\sqcup}W_{[2]}xy.$$

于是有

$$M_{\sqcup}=M_{\sqcup}W_{[2]},$$

即

$$\delta_2[\alpha\ \beta\ \delta\ \gamma]=\delta_2[\alpha\ \delta\ \beta\ \gamma].$$

故 $\beta=\delta$. 于是

$$M_{\sqcup}=\delta_2[\alpha\ \beta\ \beta\ \gamma].$$

同理

$$M_{\sqcap}=\delta_2[a\ b\ b\ c].$$

下面考虑结合律. 由

$$(x\sqcup y)\sqcup z=x\sqcup(y\sqcup z)$$

有

$$M_{\sqcup}^2xyz=M_{\sqcup}xM_{\sqcup}yz=M_{\sqcup}\left(I_2\otimes M_{\sqcup}\right)xyz.$$

于是

$$M_{\sqcup}^2=M_{\sqcup}\left(I_2\otimes M_{\sqcup}\right). \tag{5.3.1}$$

展开再比较系数可得 (5.3.1) 成立, 当且仅当

$$(1-\alpha)\gamma=(1-\beta)\beta,$$

当且仅当

$$\gamma\neq 1,\quad 如果\ \ \alpha=0. \tag{5.3.2}$$

下面考虑归一律:

$$0\sqcup x=x.$$

即

$$M_{\sqcup}\delta_2^2x=x.$$

于是有

$$M_{\sqcup}\delta_2^2=I_2.$$

即

$$M_{\sqcup}=\delta_2[\alpha\ 1\ 1\ 2]. \tag{5.3.3}$$

类此, 由

$$1\sqcap x=x$$

可得

$$M_{\sqcap} = \delta_2[1\ 2\ 2\ c]. \tag{5.3.4}$$

再考虑分配律. 由

$$x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z)$$

可得

$$M_{\sqcup}\left(I_2 \otimes M_{\sqcap}\right) = M_{\sqcap} M_{\sqcup}\left(I_4 \otimes M_{\sqcup}\right)\left(I_2 \otimes W_{[2]}\right) M_r. \tag{5.3.5}$$

由

$$x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$$

可得

$$M_{\sqcap}\left(I_2 \otimes M_{\sqcup}\right) = M_{\sqcup} M_{\sqcap}\left(I_4 \otimes M_{\sqcap}\right)\left(I_2 \otimes W_{[2]}\right) M_r. \tag{5.3.6}$$

将 (5.3.3) 及 (5.3.4) 代入 (5.3.5) 与 (5.3.6), 最后得到的容许解是 $\alpha = 1$, $c = 2$.

于是有

$$M_{\sqcup} = \delta_2[1\ 1\ 1\ 2], \tag{5.3.7}$$

$$M_{\sqcup} = \delta_2[1\ 2\ 2\ 2]. \tag{5.3.8}$$

最后检验补律. 由

$$x \sqcup x' = 1$$

可得

$$M_{\sqcup} x M_{'} x = M_{\sqcup}\left(I_2 \otimes M_{'}\right) M_r x = \delta_2^1.$$

$$\begin{bmatrix} 1 & q \\ 0 & 1-q \end{bmatrix} x = \delta_2^1.$$

取 $x = \delta_2^2$ 可知 $q = 1$. 由

$$x \sqcap x' = 0$$

可得

$$M_{\sqcap} x M_{'} x = M_{\sqcap}\left(I_2 \otimes M_{'}\right) M_r x = \delta_2^2.$$

$$\begin{bmatrix} p & 0 \\ 1-p & 1 \end{bmatrix} x = \delta_2^2.$$

取 $x=\delta_2^1$ 可知 $p=1$. 因此

$$M_{'} = \delta_2[2\ 1]. \tag{5.3.9}$$

综上可知

定理 5.3.1 任何一个二元布尔代数同构于由逻辑 $(\mathcal{D}=\{1,0\},\vee,\wedge,\neg,1,0)$ 生成的布尔代数.

5.4 注释与参考

布尔代数的基本定律首次发表于 1854 年乔治 • 布尔的《思维规律研究》一书中[29]. 由于布尔函数只定义在 $\{0,1\}$ 上, 它比起许多纯数学领域来, 似乎简单许多. 然而, 在数学上通常是简单的东西更有生命力, 即所谓 "大道从简". 布尔函数在逻辑、线路设计、密码学以及计算机科学等学科上起着无可取代的作用. 布尔出身贫寒, 他小学毕业就辍学了, 从此全靠自学, 成就了一番事业. 在《数学指南》一书中, 他被称为 "符号逻辑之父和计算机科学的奠基者之一"[81].

第6章　布尔矩阵

6.1　布尔向量空间

定义 6.1.1　给定布尔代数 B.

1) 集合

$$V_n = \left\{(a_1\ a_2\ \cdots\ a_n)\middle| a_i \in B,\ i = 1, \cdots, n\right\}$$

中的元素称为 n 维布尔向量.

2) 在 V_n 上定义加法:

$$(a_1\ \cdots\ a_n) + (b_1\ \cdots\ b_n) := (a_1 + b_1\ \cdots\ a_n + b_n),$$

这里, 每个分量的加法是布尔代数中的加法. V_n 加上其上的这种加法称为 n 维布尔向量空间.

设 $a = (a_1\ a_2\ \cdots\ a_n) \in V_n$, 它的补向量定义为 $a^c = (a_1^c\ a_2^c\ \cdots\ a_n^c)$. 另外, 我们把零向量 $\mathbf{0}$ 简记为 0.

定义 6.1.2　1) 设 $H \subset V_n$, $0 \in H$, 且如果 $a, b \in H$, 则 $a + b \in H$. 则 H 称为 V_n 的一个子空间.

2) 设 $W \subset V_n$, 包含 W 的所有子空间的交称为由 W 生成的子空间, 记作 $\langle W\rangle$. 为方便计, 设空和为 0. 这样, $\langle W\rangle$ 就是 W 的元素的所有有限和集合.

例 6.1.1　1) 令

$$W = \{(1\ 0\ 0), (0\ 1\ 0), (0\ 0\ 1)\}.$$

则

$$\begin{aligned}\langle W\rangle = \{&(0\ 0\ 0),\ (0\ 0\ 1),\ (0\ 1\ 0),\ (0\ 1\ 1)\\ &(1\ 0\ 0), (1\ 0\ 1), (1\ 1\ 0), (1\ 1\ 1)\}.\end{aligned}$$

它是 V_3 的子空间. 当 $B = \{0, 1\}$ 时, 它就是 V_3.

2) 令

$$W = \{(1\ 0\ 0), (1\ 1\ 0), (1\ 1\ 1)\}.$$

则

$$\langle W\rangle = \{(0\ 0\ 0), (1\ 0\ 0), (1\ 1\ 0), (1\ 1\ 1)\}.$$

定义 6.1.3 设 $W \subset V_n$.

1) 称 $v \in V_n$ 与 W 无关, 如果 $v \notin \langle W \rangle$.

2) W 称为无关子集, 如果对任一 $w \in W$, w 与 $W \backslash \{w\}$ 无关.

例 6.1.2 设 $W = \{(1\,0\,0), (1\,1\,0)\}$. 容易验证: W 是无关集. 设 $v = (0\,1\,0)$. 不难证明 v 与 W 无关. (注意: 它与普通向量相关性不同.)

定义 6.1.4 设 $W \subset V_n$ 为一子空间. 如果 B 是无关子集, 且 $W = \langle B \rangle$, 则称 $B \subset V_n$ 为 W 的基底.

定理 6.1.1 设 $W \subset V_n$ 为一子空间. 则存在它的一个基底 B. 并且, 该基底唯一.

证明 记 B 为 W 中的原子集. 我们证明: $W = \langle B \rangle$. 反证, 设 $W \backslash \langle B \rangle \neq \varnothing$, 则存在 $v \in W \backslash \langle B \rangle$ 为该子集的原子. 但 $v \notin \langle B \rangle$, 故 $v = v_1 + v_2$. 但是, 必然有 $v_1, v_2 \in \langle B \rangle$, 因为否则, 不妨设 $v_1 \notin \langle B \rangle$. 那么, 存在 $v_1 \in W \backslash \langle B \rangle$, 使 $0 < v_1 < v$, 与 v 为 $W \backslash \langle B \rangle$ 中的原子矛盾. 于是 $v = v_1 + v_2 \in \langle B \rangle$, 矛盾. 这说明 $W \backslash \langle B \rangle = \varnothing$.

下面证唯一性. 设 B' 是另一组基底. 那么 $B \subset B'$. 否则, 存在 $0 \neq v \in B \backslash B'$. $v = v_1 + \cdots + v_s$, $v_k \in B'$, $k = 1, \cdots, s$, 但 $v \in B$ 是原子, 矛盾. 现在设 $B' \backslash B \neq \varnothing$, 则有 $v \in B' \backslash B$ 为 $B' \backslash B$ 上的原子. 仿前半部证明可证, v 与 B 相关, 矛盾. 故 $B = B'$. □

6.2 布尔矩阵

给定一个布尔代数 B. 一个 $m \times n$ 维布尔矩阵 $A = (a_{ij})$, 这里 $a_{ij} \in B$. $m \times n$ 维布尔矩阵集合记作 $\mathcal{B}_{m \times n}$. 我们用普通矩阵乘法规则来定义布尔矩阵的乘积, 只是其中涉及的 "+" 和 "×" 均为布尔代数中的加和乘. 因为单位阵可以看作任何布尔代数下的布尔矩阵, 两个布尔矩阵的半张量积可依定义 1.2.1 来实现.

例 6.2.1 设 $B = \{0, 1\}$.

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad V = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

那么

$$A + U = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix},$$

$$A \times V = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix},$$

$$
\begin{aligned}
A \ltimes U &= (A \otimes I_2)(B \otimes I_3) \\
&= \begin{bmatrix} 0&0&0&0&1&0&1&0&0\\ 0&0&0&0&0&1&0&1&0\\ 0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&1&0&0&0 \end{bmatrix}.
\end{aligned}
$$

例 6.2.2　考虑集合运算布尔代数. 设 $B = P(N)$, 这里 $N = \{0,1,2\}$, $\sqcup = \cup$, $\sqcap = \cap$, $' = {}^c$. 那么

$$P(N) = \{\varnothing, \{0\}, \{1\}, \{2\}, \{0,1\}, \{0,2\}, \{1,2\}, \{0,1,2\}\}.$$

记 $(1) := \varnothing$, $(2) := \{0\}$, $(3) := \{1\}$, $(4) := \{2\}$, $(5) := \{0,1\}$, $(6) := \{0,2\}$, $(7) := \{1,2\}$, $(8) := \{0,1,2\}$.

设

$$U = \begin{bmatrix} (1) & (2) & (5) \\ (3) & (6) & (8) \end{bmatrix}, \quad V = \begin{bmatrix} (2) & (4) \\ (7) & (5) \\ (8) & (6) \end{bmatrix}.$$

那么

$$
\begin{aligned}
U + V^{\mathrm{T}} &= \begin{bmatrix} (1)+(2) & (2)+(7) & (5)+(8) \\ (3)+(4) & (6)+(5) & (8)+(6) \end{bmatrix} \\
&= \begin{bmatrix} (2) & (8) & (8) \\ (7) & (8) & (8) \end{bmatrix}, \\
UV &= \begin{bmatrix} (5) & (2) \\ (8) & (6) \end{bmatrix}.
\end{aligned}
$$

在此后的讨论中, 假定 $B = \mathcal{D}$. 下面讨论一个布尔矩阵的行空间与列空间.

定义 6.2.1　给定布尔矩阵 $A \in \mathcal{B}_{m\times n}$. 它的行空间是由它的行张成的 V_n 的子空间, 记作

$$\mathcal{R}(A) = \langle \mathrm{Row}(A) \rangle.$$

它的列空间是由它的列张成的 V_m 的子空间, 记作

$$\mathcal{C}(A) = \langle \mathrm{Col}(A) \rangle.$$

根据乘法规则, 显然有以下结果:

命题 6.2.1　设 $A \in \mathcal{B}_{m\times n}$, $B \in \mathcal{B}_{n\times t}$. 那么

$$\mathcal{R}(AB) \subset \mathcal{R}(B); \quad \mathcal{C}(AB) \subset \mathcal{C}(A).$$

例 6.2.3 设

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}.$$

那么, 不难算出

$$\begin{aligned}\mathcal{C}(A) = \{&(0,0,0),(0,0,1),(0,1,0),(1,0,0)\\&(0,1,1),(1,0,1),(1,1,0),(1,1,1)\},\end{aligned}$$

$$\begin{aligned}\mathcal{R}(A) = \{&(0,0,0,0),(1,0,0,0),(0,1,0,1),(0,1,1,0)\\&(0,1,1,1),(1,1,0,1),(1,1,1,0),(1,1,1,1)\}.\end{aligned}$$

从上例看, 有 $|\mathcal{C}(A)| = |\mathcal{R}(A)| = 8$. 这个结论一般都对. 这类似但又不同于一般矩阵的一个性质: 行秩与列秩相等.

定理 6.2.1 设 $A \in \mathcal{B}_{m\times n}$. 那么

$$|\mathcal{R}(A)| = |\mathcal{C}(A)|. \tag{6.2.1}$$

证明 记 $\xi = \{1, 2, \cdots, m\}$, 构造一个映射 $\pi : \mathcal{C}(A) \to \mathcal{R}(A)$ 如下: 设 $v \in \mathcal{C}(A)$, 则存在 $\alpha \subset \xi$, 使得

$$v = \sum_{i\in\alpha} \delta_m^i.$$

记 $\alpha^c = \xi\backslash\alpha$. 定义

$$\pi(v) = \sum_{i\in\alpha^c} \mathrm{Row}_i(A).$$

先证 π 是映上的. 设 $w \in \mathcal{R}(A)$, 则存在 $\beta \subset \xi$, 使 $w = \sum\limits_{i\in\beta} \mathrm{Row}_i(A)$. 令 $\alpha = \beta^c$, $v = \sum\limits_{i\in\alpha} \delta_m^i$. 则 $\pi(v) = w$.

再证 π 是一对一的. 用归谬法, 设 $\alpha \subset \xi$, $\beta \subset \xi$,

$$u = \sum_{i\in\alpha} \delta_m^i, \quad v = \sum_{i\in\beta} \delta_m^i.$$

假定 $\alpha \neq \beta$, 即 $u \neq v$. 但有 $\pi(u) = \pi(v)$, 即

$$\sum_{i\in\alpha^c} \mathrm{Row}_i(A) = \sum_{i\in\beta^c} \mathrm{Row}_i(A).$$

由于 $\alpha \neq \beta$, 不妨设: 存在 $p \in \beta\backslash\alpha$. 因为 $v \in \mathcal{C}(A)$, 则存在 $k \in \xi$, $a_{pk} = 1$, 并且 $\mathrm{Col}_k(A) \leqslant v$. 因为 $p \in \alpha^c$, 则 $\pi(u) \geqslant \mathrm{Row}_p(A)$, 故 $[\pi(u)]_k = 1$. 于是, $[\pi(v)]_k = 1$. 由定义, 存在某个 $q \in \beta^c$ 使 $a_{qk} = 1$. 但已知 $\mathrm{Col}_k(A) \leqslant v$, 因此, $\delta_m^q \leqslant v$. 但这不可能, 因为 $q \in \beta^c$. 故 $\pi(u) \neq \pi(v)$. □

推论 6.2.1 $|\mathcal{C}(AB)| \leqslant |\mathcal{C}(A)|$, 且 $|\mathcal{R}(AB)| \leqslant |\mathcal{R}(B)|$.

推论 6.2.2 设 A 和 π 同定理 6.2.1. $u, v \in \mathcal{C}(A)$. 则 $u \leqslant v$, 当且仅当, $\pi(u) \geqslant \pi(v)$.

证明 使用定理 6.2.1 的证明中的记号. 必要性是显然的, 因为 $\alpha \subset \beta$, 则 $\beta^c \subset \alpha^c$. 至于充分性, 设 $\pi(u) \leqslant \pi(v)$, 但反设 $u \not\leqslant v$. 取 $p \in \alpha \backslash \beta$. 利用与定理 6.2.1 的证明中相同的讨论可导出矛盾. □

定义 6.2.2 给定 $A \in \mathcal{B}_{m\times n}$. 记 $B_r(A)$ ($B_c(A)$) 为 A 的行 (列) 空间的基底. 定义 A 的行 (列) 秩为

$$\rho_r(A) = |B_r(A)|, \quad (\rho_c(A) = |B_c(A)|).$$

与普通矩阵秩不同, 布尔矩阵的行秩与列秩未必相等.

例 6.2.4 考虑

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

不难验证, $\rho_r(A) = 4$, $\rho_c(A) = 3$.

6.3 检测问题

检测问题可以看做布尔矩阵的一个应用, 具体描述如下: 假定某患者的可能疾病有 $d_1, \cdots, d_n$, 化验手段有 $t_1, \cdots, t_m$. 如果化验手段 t_i 对疾病 d_j 有效 (呈阳性), 则令 $a_{i,j} = 1$, 否则令 $a_{ij} = 0$. 这样, 就可以得到一个矩阵 $A = (a_{ij}) \in \mathcal{B}_{m\times n}$, 称为检测矩阵. 这类问题很多, 例如 d_j 可以是矿石可能含有的元素, t_i 为试剂, 检测的目的是测定矿石是否含元素 d_j.

对于检测矩阵, 一个典型问题是 Ledley 问题[91]: 要确诊某患者是否患有疾病 d_i, 准确诊断所需要的最少检测数? 下面给出一个算法.

算法 6.3.1 • 第一步: 将所检测的列 d_i 中的每个零元素 a_{ji} 所在行 j 中的 t_j 改为 t_j^c, 然后将 a_{js} 改为 $\neg a_{js}$, $s = 1, \cdots, n$.

• 第二步: 寻找零最多的行, α, 做 t_α 检验. 然后检查 $a_{\alpha,\beta}$ $(\forall \beta)$. 如果 $a_{\alpha,\beta} = 0$ 则将 β 列删去. 最后删除第 α 行.

• 第三步: 对剩余的矩阵重复第二步, 直到 (i) 情况 1: 只剩检测列 d_i, 则历次所得 t_α 构成有效检测序列. (ii) 情况 2: 剩余多于一列 (检测列) 但已无带零的行, 则检测无法唯一确定疾病.

注 1) 算法 6.3.1 的合理性在于, 它在每一步出局的疾病数都是最多的, 因此, 它有较少的检验数. 但这仍然不能保证这是最少检验数的, 因为不同路径的每一步出局疾病数不能相比.

2) 如果情况 2 发生, 那说明留下的每一列 (每一种病) 都是有可能的, 但已删去的就不可能了.

用一个例子来说明算法 6.3.1.

例 6.3.1 ([82]) 设检测矩阵 A 如下 (表 6.3.1), 找出尽可能少的检测数确定 d_1.

表 6.3.1 检测表 -1

	d_1	d_2	d_3	d_4	d_5
t_1	1	0	0	1	0
t_2	0	1	1	0	0
t_3	1	0	0	0	1
t_4	1	1	0	0	1

第一步, 在本例中, 检测的列为 d_1. 该列中有一个零, 为 $a_{2,1}$. 所以将第二行取"非". 注意: 在第一步完成后, 第 d_1 列元素全部为 1(表 6.3.2).

表 6.3.2 检测表 -2

	d_1	d_2	d_3	d_4	d_5
t_1	1	0	0	1	0
$\neg t_2$	1	0	0	1	1
t_3	1	0	0	0	1
t_4	1	1	0	0	1

第二步, 取 $\alpha = 1$, (由于第 1 行零个数为 3), 做检验 t_1. 然后删去所有 $a_{1,\beta} = 0$ 的列 β. 即, $\beta = 2, 3, 5$. 再删去第 1 行. 剩下的矩阵见表 6.3.3.

表 6.3.3 检测表 -3

	d_1	d_4
$\neg t_2$	1	1
t_3	1	0
t_4	1	0

第三步, 对矩阵 6.3.3 选 $\alpha = 2$ 或 $\alpha = 3$, 即, 做检验 t_3 或 t_4. 然后删去 d_4, 仅余 d_1.

结论: $\{t_1, t_3\}$ 或 $\{t_1, t_4\}$ 均为有效检验.

当然, 在第二步, 也可取 $\alpha = 3$, (由于第 3 行零个数也为 3), 做检验 t_3. 然后删

去所有 $a_{3,\beta}=0$ 的列 β. 即, $\beta=2,3,4$. 再删去第 3 行. 剩下的矩阵见表 6.3.4.

表 6.3.4　检测表 -4

	d_1	d_5
t_1	1	0
$\neg t_2$	1	1
t_4	1	1

第三步, 对矩阵 6.3.3 选 $\alpha=1$, 即, 做检验 t_1. 结论: $\{t_3,\ t_1\}$ 也是有效检验.

下面讨论可检测问题: 即给了一张检测表, 问何时 d_j 是可检测的. 下面这个结论很简单.

命题 6.3.1　给定一个检测表 $A=(a_{ij})\in\mathcal{B}_{m\times n}$.

1) d_j 是可检测的, 当且仅当

$$\mathrm{Col}_s(A)\neq\mathrm{Col}_j(A),\quad s\neq j.$$

2) $d_j,\ j=1,\cdots,n$ 可检测, 当且仅当

$$\mathrm{Col}_p(A)\neq\mathrm{Col}_q(A),\quad 1\leqslant p<q\leqslant n.$$

6.4　逻辑关系方程

考察一组逻辑关系式

$$\varphi_i(\xi_1,\cdots,\xi_n)=c_i,\quad i=1,\cdots,s, \tag{6.4.1}$$

这里 $c_i\in\mathcal{D}$ 为常值.

定义 6.4.1　1) 逻辑关系式 (6.4.1) 的前提解 (antecedence solution) 是指关于 $\{\xi_1,\cdots,\xi_n\}$ 的逻辑关系, 它保证 (6.4.1) 成立.

2) 逻辑关系式 (6.4.1) 的推论解 (consequence solution) 是指当 (6.4.1) 成立时, 强迫 $\{\xi_1,\cdots,\xi_r\}$ 满足的逻辑关系.

文献 [97] 将逻辑变量分为两组, 即

$$\{\xi_1,\cdots,\xi_n\}=\{x_1,\cdots,x_p\}\bigcup\{y_1,\cdots,y_q\},$$

然后, 寻找形如

$$y_\ell=f_\ell(x_1,\cdots,x_p),\quad \ell=1,\cdots,q$$

的解. 为方便计, 称其为 Ledley 型解 (包括 Ledley 型前提解、Ledley 型推论解). 以下只讨论 Ledley 型解. 这时, (6.4.1) 变为

$$\varphi_i(x_1,\cdots,x_p;y_1,\cdots,y_q)=c_i,\quad i=1,\cdots,s. \tag{6.4.2}$$

定义 6.4.2 考虑逻辑表达式 (6.4.1). 将自变量用向量表示, 并记 $x=\ltimes_{i=1}^{p}x_i$, $y=\ltimes_{i=1}^{q}y_i$. (6.4.1) 的真值矩阵, 记为 $T_{\varphi}^{(X,Y)}\in\mathcal{B}_{2^q\times 2^p}$ 定义如下:

$$(T_{\varphi}^{(X,Y)})_{i,j}=\begin{cases}1, & \varphi_\mu\left(y=\delta_{2^q}^{i},x=\delta_{2^p}^{j}\right)=c_\mu,\forall\mu,\\ 0, & \text{其他}.\end{cases} \tag{6.4.3}$$

我们用一个例子来说明真值矩阵.

例 6.4.1 考虑逻辑表达式

$$\Sigma:\ \varphi(\xi_1,\xi_2,\xi_3)=(\xi_1\wedge\xi_2)\vee(\neg\xi_3)=1. \tag{6.4.4}$$

约定分割为

$$\Xi=\{\xi_1,\xi_2,\xi_3\}=X\cup Y, \tag{6.4.5}$$

其中 $X:=\{\xi_1,\xi_2\}$ 和 $Y=\{\xi_3\}$. 那么, 不难得到表 6.4.1.

表 6.4.1 (6.4.4) 的真值矩阵

$y\backslash x$	1, 1	1, 0	0, 1	0, 0
1	1	0	0	0
0	1	1	1	1

因此,

1) 如果 $X=\{\xi_1,\xi_2\}$, $Y=\{\xi_3\}$, 那么, 有真值矩阵:

$$T_1=\begin{bmatrix}1&0&0&0\\1&1&1&1\end{bmatrix};$$

类似地,

2) 如果 $X=\{\xi_2,\xi_3\}$, $Y=\{\xi_1\}$, 那么,

$$T_2=\begin{bmatrix}1&1&0&1\\0&1&0&1\end{bmatrix};$$

3) 如果 $X=\{\xi_1,\xi_3\}$, $Y=\{\xi_2\}$, 那么,

$$T_3=\begin{bmatrix}1&1&0&1\\0&1&0&1\end{bmatrix};$$

4) 如果 $X=\{\xi_1\}, Y=\{\xi_2,\xi_3\}$, 那么,

$$T_4=\begin{bmatrix}1&0\\1&1\\0&0\\1&1\end{bmatrix};$$

5) 如果 $X=\{\xi_2\}, Y=\{\xi_1,\xi_3\}$, 那么,

$$T_5=\begin{bmatrix}1&0\\1&1\\0&0\\1&1\end{bmatrix};$$

6) 如果 $X=\{\xi_3\}, Y=\{\xi_1,\xi_2\}$, 那么,

$$T_6=\begin{bmatrix}1&1\\0&1\\0&1\\0&1\end{bmatrix}.$$

6.5 逻辑关系方程的 Ledley 解

实际上, 真值矩阵包括了逻辑关系方程的所有信息, 而解即可从中得出. 我们从分析一个例子开始.

例 6.5.1 ([91]) 考察以下逻辑方程

$$\varphi(\xi_1,\xi_2,\xi_3)=(\neg\xi_1\wedge\xi_3)\vee(\xi_1\wedge\xi_2\wedge\neg\xi_3)=0. \tag{6.5.1}$$

设分割为: $X=\{\xi_1,\xi_2\}, Y=\{\xi_3\}$. 则 (6.5.1) 相应的真值矩阵见表 6.5.1.

表 6.5.1 (6.5.1) 的真值矩阵

$\xi_3\backslash\xi_1\xi_2$	δ_4^1	δ_4^2	δ_4^3	δ_4^4
δ_2^1	1	1	0	0
δ_2^2	0	1	1	1

写成矩阵形式, 则有

$$T_{(6.5.1)}^{(X,Y)}=\begin{bmatrix}1&1&0&0\\0&1&1&1\end{bmatrix}. \tag{6.5.2}$$

找一个逻辑矩阵 $M \in \mathcal{L}_{2\times 4}$, 使得

$$M \leqslant T^{(X,Y)}_{(6.5.1)}. \tag{6.5.3}$$

这里, 对矩阵 $A = (a_{i,j})$, $B = (b_{i,j}) \in \mathcal{M}_{m\times n}$, $A \leqslant B$ 表示

$$a_{i,j} \leqslant b_{i,j}, \quad i = 1, \cdots, m;\ j = 1, \cdots, n.$$

令

$$\xi_3 = M\xi_1\xi_2. \tag{6.5.4}$$

那么, (6.5.4) 就是 (6.5.1) 的一个前提解. 这是因为, 对每个 $x = \xi_1\xi_2$ 以及相应的 $y = \xi_3$, 由于 (6.5.3) 不难看出 (6.5.1) 总成立. 因此, (6.5.4)→ (6.5.1). 反之, (6.5.3) 也是 (6.5.4) 蕴涵 (6.5.1) 的必要条件. 否则, 至少有一个 $(\xi_1, \xi_2) = (\alpha, \beta)$ 和 $\xi_3 = M\alpha\beta$, 它们对应一个位置 (i, j), 使得 $m_{i,j} > t_{i,j}$ (即在 (i, j) 点有 M 为 1 而 T 为 0). 那么, 当 $(\xi_1, \xi_2) = (\alpha, \beta)$ 且 $\xi_3 = M\alpha\beta$ 时, (6.5.1) 不成立.

回到原系统, 则显然有两个 M 满足 (6.5.3), 它们是

$$M_1 = \delta_2[1\ 1\ 2\ 2]; \quad M_2 = \delta_2[1\ 2\ 2\ 2].$$

根据以上讨论可知: 对应于以上的分割, (6.5.1) 有两个 Ledley 型前提解, 它们是

$$\xi_3 = \delta_2[1\ 1\ 2\ 2]\xi_1\xi_2 \tag{6.5.5}$$

和

$$\xi_3 = \delta_2[1\ 2\ 2\ 2]\xi_1\xi_2. \tag{6.5.6}$$

为讨论一般情况, 回忆汉明权重: 设 $X = (x_1, \cdots, x_n) \in \mathcal{B}_n$, 则其汉明权重为[37]

$$w_H(X) = \sum_{i=1}^{n} x_i.$$

根据对例 6.5.1 的讨论, 不难得到如下结果:

定理 6.5.1 考察逻辑系统 (6.4.1).

1) 设 (X, Y) 为未知变量 $\Xi = \{\xi_1, \cdots, \xi_n\}$ 的一组分割. $|X| = p > 0$, $|Y| = q > 0$ $(p + q = n)$. 对应于分割 (X, Y) 的真值矩阵为 $T := T^{(X,Y)}_{\varphi} \in \mathcal{B}_{2^q\times 2^p}$. 那么, (6.4.1) 有 Ledley 前提解, 当且仅当, T 的每一列的汉明权重均非零, 即

$$w_j := w_H(\mathrm{Col}_j(T)) > 0, \quad j = 1, \cdots, 2^p. \tag{6.5.7}$$

2) 设逻辑矩阵 $M \in \mathcal{L}_{2^q \times 2^p}$ 满足 $M \leqslant T$, 则 $y = Mx$ 为 (6.4.1) 的前提解.

3) 对应于给定的分割 (X, Y), 有

$$r = \prod_{j=1}^{2^p} w_j \tag{6.5.8}$$

个前提解.

4) 设 (X_i, Y_i), $i = 1, \cdots, \ell$ 为未知变量集合 $\varXi$ 的所有分割, $|X_i| = p_i$. 那么, (6.4.1) 的前提解个数为

$$s_a \leqslant \sum_{i=1}^{\ell} \prod_{j=1}^{2^{p_i}} w_H \left(\mathrm{Col}_j \left(T_{\varphi}^{(X_i, Y_i)} \right) \right). \tag{6.5.9}$$

注　1) 这里, 分割 (X, Y) 是有序的, 即

$$(X, Y) \neq (Y, X).$$

这是因为分割的两个部分起的作用是不一样的. 前者为自变量, 后者为因变量.

2) (6.5.9) 中之所以出现 "小等于 $(\leqslant)$" 号, 是因为不同分割可能出现相同的前提解.

例 6.5.2　回忆例 6.4.1. 那里有 6 个不同的分割, 对应于 6 个不同的真值矩阵 T_i, $i = 1, \cdots, 6$.

1) 对应于 T_1, (6.4.4) 有 2 个前提解:

$$\xi_3 = \delta_2[1, 2, 2, 2]\xi_1\xi_2 = \xi_1 \wedge \xi_2.$$

$$\xi_3 = \delta_2[2, 2, 2, 2]\xi_1\xi_2 = \delta_2^2.$$

2) 对应于 T_2, 因为 $w_H\left(\mathrm{Col}_3(T_2)\right) = 0$, 所以 (6.4.4) 没有前提解.

3) 对应于 T_3, 因为 $w_H\left(\mathrm{Col}_3(T_2)\right) = 0$, 所以 (6.4.4) 没有前提解.

4) 对应于 T_4, (6.4.4) 有 6 个前提解:

- 情形 1, $\xi_2\xi_3 = \delta_4[1, 2]\xi_1$:

　　则有

$$\xi_2 = \left(I_2 \otimes \mathbf{1}_2^{\mathrm{T}}\right) \delta_4[1, 2]\xi_1 = \delta_2[1, 1]\xi_1 = \delta_2^1.$$

$$\xi_3 = \left(\mathbf{1}_2^{\mathrm{T}} \otimes I_2\right) \delta_4[1, 2]\xi_1 = \delta_2[1, 2]\xi_1 = \xi_1.$$

于是有解

$$\begin{cases} \xi_2 = \delta_2^1, \\ \xi_3 = \xi_1. \end{cases}$$

同理, 有

- 情形 2, $\xi_2\xi_3 = \delta_4[1,4]\xi_1$: 此时

$$\begin{cases} \xi_2 = \xi_1, \\ \xi_3 = \xi_1. \end{cases}$$

- 情形 3, $\xi_2\xi_3 = \delta_4[2,2]\xi_1$: 此时

$$\begin{cases} \xi_2 = \delta_2^1, \\ \xi_3 = \delta_2^2. \end{cases}$$

- 情形 4, $\xi_2\xi_3 = \delta_4[2,4]\xi_1$: 此时

$$\begin{cases} \xi_2 = \xi_1, \\ \xi_3 = \delta_2^2. \end{cases}$$

- 情形 5, $\xi_2\xi_3 = \delta_4[4,2]\xi_1$: 此时

$$\begin{cases} \xi_2 = \neg\xi_1, \\ \xi_3 = \delta_2^2. \end{cases}$$

- 情形 6, $\xi_2\xi_3 = \delta_4[4,4]\xi_1$: 此时

$$\begin{cases} \xi_2 = \delta_2^2, \\ \xi_3 = \delta_2^2. \end{cases}$$

5) 对应于 T_5, (6.4.4) 有 6 个前提解:

$$\begin{cases} \xi_1 = \delta_2^1, \\ \xi_3 = \xi_2; \end{cases} \quad \begin{cases} \xi_1 = \xi_2, \\ \xi_3 = \xi_2; \end{cases} \quad \begin{cases} \xi_1 = \delta_2^1, \\ \xi_3 = \delta_2^2; \end{cases}$$
$$\begin{cases} \xi_1 = \xi_2, \\ \xi_3 = \delta_2^2; \end{cases} \quad \begin{cases} \xi_1 = \neg\xi_2, \\ \xi_3 = \delta_2^2; \end{cases} \quad \begin{cases} \xi_1 = \delta_2^2, \\ \xi_3 = \delta_2^2. \end{cases}$$

6) 对应于 T_6, (6.4.4) 有 4 个前提解:

$$\begin{cases} \xi_1 = \delta_2^1, \\ \xi_2 = \delta_2^1; \end{cases} \quad \begin{cases} \xi_1 = \delta_2^1, \\ \xi_2 = \xi_3; \end{cases}$$
$$\begin{cases} \xi_1 = \xi_3, \\ \xi_2 = \delta_2^1; \end{cases} \quad \begin{cases} \xi_1 = \xi_3, \\ \xi_2 = \xi_3. \end{cases}$$

考察系统 (6.4.1), 设分割 (X,Y) 给定, 下面考虑推论解. 仔细观察真值矩阵, 类似前提解的讨论, 不难发现以下结论:

定理 6.5.2 考察逻辑系统 (6.4.1).

1) 设 (X,Y) 为未知变量 $\Xi=\{\xi_1,\cdots,\xi_n\}$ 的一组分割. $|X|=p>0$, $|Y|=q>0$ $(p+q=n)$. 对应于分割 (X,Y) 的真值矩阵为 $T:=T_{\varphi}^{(X,Y)}\in\mathcal{B}_{2^q\times 2^p}$. 那么, (6.4.1) 有 Ledley 推论解, 当且仅当, T 的每一列的汉明权重均满足

$$w_j:=w_H(\mathrm{Col}_j(T))\leqslant 1,\quad j=1,\cdots,2^p. \tag{6.5.10}$$

2) 设逻辑矩阵 $M\in\mathcal{L}_{2^q\times 2^p}$ 满足 $M\geqslant T$, 则 $y=Mx$ 为 (6.4.1) 的推论解.

3) 在给定的分割 (X,Y) 下, 设 $|X|=p$, 且 μ 为 T 中元素全为 0 的列的个数, 则 (6.4.1) 的推论解的个数为

$$s=\begin{cases}0, & (6.5.10)\text{ 不成立;}\\ 2^{\mu p}, & (6.5.10)\text{ 成立.}\end{cases} \tag{6.5.11}$$

4) 设 (X_i,Y_i), $i=1,\cdots,\ell$ 为未知变量集合 Ξ 的所有分割, 那么, (6.4.1) 的推论解的总数为

$$s_c\leqslant\sum_{i=1}^{\ell}s_i, \tag{6.5.12}$$

其中 s_i 为在给定分割 (X_i,Y_i) 下的推论解的个数.

下面给出一个例子.

例 6.5.3 考察系统

$$\varphi(\xi_1,\xi_2,\xi_3)=[\xi_1\wedge(\xi_2\bar{\vee}\xi_3)]\vee[\xi_2\vee\neg(\xi_2\vee\xi_3)]. \tag{6.5.13}$$

• 设 $X_1=\{\xi_1,\xi_2\}$, $Y_1=\{\xi_3\}$, 则在分割 (X_1,Y_1) 下相应真值矩阵为

$$T_{(6.5.13)}^{(X_1,Y_1)}=\begin{bmatrix}1&0&0&1\\0&1&0&0\end{bmatrix}. \tag{6.5.14}$$

由定理 6.5.2 知, 此时有两个推论解:

$$\xi_3=\begin{bmatrix}1&0&1&1\\0&1&0&0\end{bmatrix}\xi_1\xi_2.$$

$$\xi_3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_1 \xi_2.$$

返回逻辑形式有

$$\xi_3 = (\xi_1 \wedge \xi_2) \vee \neg \xi_1. \tag{6.5.15}$$

及

$$\xi_3 = \neg(\xi_1 \bar{\vee} \xi_2). \tag{6.5.16}$$

- 设 $X_2 = \{\xi_1, \xi_3\}$, $Y_2 = \{\xi_2\}$, 则在分割 (X_2, Y_2) 下相应真值矩阵为

$$T_{(6.5.13)}^{(X_2,Y_2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{6.5.17}$$

由定理 6.5.2 知，此时有两个推论解:

$$\xi_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_1 \xi_3.$$

$$\xi_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \xi_1 \xi_3.$$

返回逻辑形式有

$$\xi_2 = \neg(\xi_1 \bar{\vee} \xi_3). \tag{6.5.18}$$

及

$$\xi_2 = \xi_1 \wedge \xi_3. \tag{6.5.19}$$

- 设 $X_3 = \{\xi_2, \xi_3\}$, $Y_3 = \{\xi_1\}$, 则在分割 (X_3, Y_3) 下相应真值矩阵为

$$T_{(6.5.13)}^{(X_3,Y_3)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{6.5.20}$$

由定理 6.5.2 知，此时有两个推论解:

$$\xi_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \xi_2 \xi_3.$$

$$\xi_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_2 \xi_3.$$

返回逻辑形式有

$$\xi_1 = \xi_2 \vee \neg(\xi_2 \vee \xi_3). \tag{6.5.21}$$

及

$$\xi_1 = \neg(\xi_2 \bar{\vee} \xi_3). \tag{6.5.22}$$

• 注意到

$$T_\varphi^{(X,Y)} = \left(T_\varphi^{(Y,X)}\right)^{\mathrm{T}},$$

则对分割 (X, Y), 其中 $|X| = 1$, $|Y| = 2$ 的情况下, 不会存在推论解.

综上, (6.5.13) 共有六个推论解, 分别为 (6.5.15), (6.5.16), (6.5.18), (6.5.19), (6.5.21), (6.5.22).

下面这个推论是显然的.

推论 6.5.1 考察系统 (6.4.1), 设分割 (X, Y) 给定. 则对应该分割:

1) 系统既有前提解又有推论解, 当且仅当, 其真值矩阵 $T_\varphi^{(X,Y)}$ 是逻辑矩阵.

2) 这时前提解与推论解唯一且相等, 为

$$y = T_\varphi^{(X,Y)} x. \tag{6.5.23}$$

3) (6.5.23) 与原系统 (6.4.1) 等价.

证明 由定理 6.5.1 和定理 6.5.2 可知: 系统既有前提解又有推论解, 则必存在逻辑矩阵 M_1, M_2, 使得 $y = M_1 x$ 为前提解, $y = M_2 x$ 为推论解, 并且

$$M_1 \geqslant T_\varphi^{(X,Y)} \geqslant M_2. \tag{6.5.24}$$

因为 M_1, M_2 为逻辑矩阵, 若 $M_1 \geqslant M_2$, 则 $M_1 = M_2$. 于是 (6.5.23) 既是唯一的前提解, 也是唯一的推论解.

由前提解与推论解的定义可知

$$(6.5.23) \Rightarrow (6.4.1) \Rightarrow (6.5.23).$$

于是, 系统 (6.5.23) 与系统 (6.4.1) 等价. □

下面给出一个例子.

例 6.5.4 考察系统 (6.4.1), 设其在向量形式下可表示为

$$
\begin{aligned}
&\varphi_1(\xi_1,\xi_2,\xi_3,\xi_4)=\delta_2[2,1,2,2,1,2,1,2,1,1,1,2,2,1,2,1]\xi_1\xi_2\xi_3\xi_4=\delta_2^1,\\
&\varphi_2(\xi_1,\xi_2,\xi_3,\xi_4)=\delta_2[2,2,2,1,2,2,1,2,1,1,2,2,1,2,2,1]\xi_1\xi_2\xi_3\xi_4=\delta_2^2.
\end{aligned} \tag{6.5.25}
$$

我们只讨论几个典型的分割, 其他的情形读者可以自行计算.

- 情况 1: 设分割为 $X_1=\{\xi_1,\xi_2\}$ 及 $Y_1=\{\xi_3,\xi_4\}$. 令 $x=\xi_1\xi_2$, $y=\xi_3\xi_4$. 真值矩阵见表 6.5.2.

表 6.5.2 (6.5.25) 的真值矩阵

$y\backslash x$	δ_4^1	δ_4^2	δ_4^3	δ_4^4
δ_4^1	0	1	0	0
δ_4^2	1	0	0	1
δ_4^3	0	0	1	0
δ_4^4	0	0	0	0

显然, (6.5.25) 在分割 (X_1,Y_1) 下的真值矩阵是一个逻辑矩阵. 根据推论 6.5.1, (6.5.25) 有唯一的推论解, 同时也是唯一的前提解:

$$
y=T_{(6.5.25)}^{(X_1,Y_1)}x:=T_1x=\delta_4[2,1,3,1]x, \tag{6.5.26}
$$

利用分量算法可得

$$
\begin{cases}
\xi_3=\left(I_2\otimes\mathbf{1}_2^{\mathrm{T}}\right)T_{(6.5.25)}^{(X_1,Y_1)}x=\delta_2[1,1,2,1]x,\\
\xi_4=\left(\mathbf{1}_2^{\mathrm{T}}\otimes I_2\right)T_{(6.5.25)}^{(X_1,Y_1)}x=\delta_2[2,1,1,1]x.
\end{cases} \tag{6.5.27}
$$

回到逻辑形式有

$$
\begin{cases}
\xi_3=\xi_1\vee\neg\xi_2,\\
\xi_4=\xi_1\bar{\vee}\xi_2.
\end{cases} \tag{6.5.28}
$$

- 情况 2: 设分割为 $X_2=\{\xi_1,\xi_3\}$ 及 $Y_2=\{\xi_2,\xi_4\}$. 令 $x=\xi_1\xi_3$, $y=\xi_2\xi_4$.

  真值矩阵为

$$
T_2:=T_{(6.5.25)}^{(X_2,Y_2)}=\begin{bmatrix}0&0&0&1\\1&0&0&0\\1&0&0&0\\0&0&1&0\end{bmatrix}. \tag{6.5.29}
$$

根据定理 6.5.1 和 6.5.2, (6.5.25) 在分割 (X_2,Y_2) 下既没有前提解, 又没有推论解.

- 情况 3: 设分割为 $X_3=\{\xi_1\}$ 及 $Y_3=\{\xi_2,\xi_3,\xi_4\}$. 令 $x=\xi_1$, $y=\xi_2\xi_3\xi_4$. 真值矩阵为

$$T_3 := T_{(6.5.25)}^{(X_3,Y_3)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{6.5.30}$$

根据定理 6.5.1 和 6.5.2, (6.5.25) 在分割 (X_3,Y_3) 下有 4 个前提解, 没有推论解.

- 情况 4: 设分割为 $X_4=\{\xi_1,\xi_2,\xi_3\}$ 及 $Y_4=\{\xi_4\}$. 令 $x=\xi_1\xi_2\xi_3$, $y=\xi_2\xi_3\xi_4$. 真值矩阵为

$$T_4 := T_{(6.5.25)}^{(X_4,Y_4)} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{6.5.31}$$

根据定理 6.5.1 和 6.5.2, (6.5.25) 在分割 (X_4,Y_4) 下没有前提解, 有 16 个推论解.

6.6 注释与参考

文献 [91] 是关于布尔矩阵的一本经典参考书, 内容丰富. 作者在前言中提到:"对于物理科学, 生物科学以及社会科学中出现的许多类离散结构模型的建模和分析, 布尔矩阵理论的应用具有本质的重要性." 其实, 他所谓 "离散结构模型" 就是本书讨论的有限集上的静态或动态模型. 布尔矩阵的确有许多应用, 例如文献 [97] 将其用于酶化学中的复杂逻辑问题, 并讨论了其对遗传密码的应用. 文献 [151] 讨论了布尔矩阵在有限自动机方面的应用等.

本章讨论的主要是布尔矩阵的基本概念与初步应用. 进一步的内容可参见文献 [91], 关于解的部分可见文献 [121]. 即使从今天的观点看, 布尔矩阵理论中仍有许多待完善的地方.

第 7 章　逻辑函数的复合分解

7.1　复 合 分 解

设 $f: \mathcal{D}^n \to \mathcal{D}$ 为一布尔函数. 在工程上它可以用一个逻辑线路来实现 (图 7.1.1(a)). 所谓复合分解是指将其转化为两个简单函数的复合. 严格定义如下.

定义 7.1.1　1) 设 $\{X_1, X_2\}$ 是 $X=\{x_1, x_2, \cdots, x_n\}$ 的一个分割. 如果存在两个布尔函数 ϕ 和 ψ, 使得 f 可表示为

$$f(X)=F(\phi(X_1), \psi(X_2)), \tag{7.1.1}$$

则称 (7.1.1) 为 f 的一个不相交复合 (2) 分解 (图 7.1.1 (b)).

2) 设 $\{X_1, X_2, X_3\}$ 是 $X=\{x_1, x_2, \cdots, x_n\}$ 的一个分割. 如果存在两个布尔函数 ϕ 和 ψ, 使得 f 可表示为

$$f(X)=F(\phi(X_1, X_2), \psi(X_2, X_3)), \tag{7.1.2}$$

则称 (7.1.2) 为 f 的一个相交复合 (2) 分解 (图 7.1.1(c)).

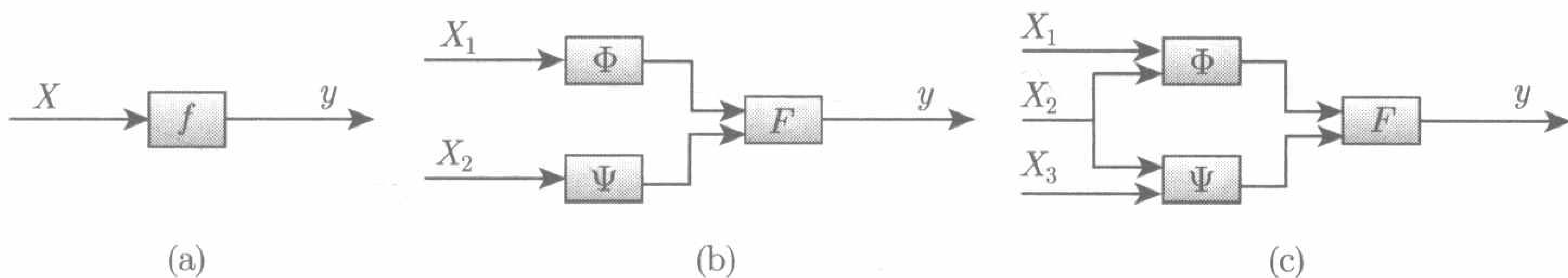


图 7.1.1　复合分解

布尔函数的复合分解在逻辑线路设计中尤为重要. 但其实, 从理论上将上述定义推广到更一般的情况是很自然的. 例如

1) 设 $f(x)$, $x_i \in \mathcal{D}_k$, $i=1, \cdots, n$, 上述定义变为 k 值逻辑函数的复合分解.

2) 设 $f(x)$, ϕ, $\psi \in \mathcal{D}_{k_0}$, $x_i \in \mathcal{D}_{k_i}$, $i=1, \cdots, n$, 上述定义变为多值逻辑函数的复合分解.

这些推广不仅有理论意义, 也有许多实际应用或应用前景.

在以下的讨论中, 我们需要两个假设:

(i) 分割的良序假设: 即, 对不相交分解

$$X_1=\{x_1, \cdots, x_r\}, \quad X_2=\{x_{r+1}, \cdots, x_n\};$$

对相交分解

$$X_1 = \{1, 2, \cdots, r_1\},$$

$$X_2 = \{r_1 + 1, r_1 + 2, \cdots, r_1 + r_2\},$$

$$X_3 = \{r_1 + r_2 + 1, r_1 + r_2 + 2, \cdots, n\}.$$

任意分割的情况可通过变量换位解决.

(ii) 复合函数的一致性:

$$f(x),\ \phi,\ \psi \in \mathcal{D}_{k_0}.$$

更一般的情况应当是 $f(x) \in \mathcal{D}_\alpha$, $\phi \in \mathcal{D}_\beta$, $\psi \in \mathcal{D}_\gamma$. 本章不讨论这种情况. 但以下的分析方法不难推广到这种情况.

7.2 不相交复合分解

定义 7.2.1 设 $f : \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 为一多值逻辑函数, $\Gamma \cup \Lambda$ 为 $\{1, 2, \cdots, n\}$ 的一个分割. f 称具有对分割 $\Gamma \cup \Lambda$ 的不相交复合分解, 如果存在 k_0 值逻辑函数 $F : \mathcal{D}_{k_0}^2 \to \mathcal{D}_{k_0}$, 和两个多值逻辑函数 $\phi : \prod_{i\in\Gamma} \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 与 $\psi : \prod_{i\in\Lambda} \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$, 使得

$$f(x_1, \cdots, x_n) = F(\phi(x_\gamma), \psi(x_\lambda)), \tag{7.2.1}$$

这里 $x_\gamma \in \prod_{i\in\Gamma} \mathcal{D}_{k_i}$, $x_\lambda \in \prod_{i\in\Lambda} \mathcal{D}_{k_i}$.

设分割是良序的, 即

$$\Gamma = \{1, 2, \cdots, r\}, \quad \Lambda = \{r + 1, r + 2, \cdots, n\}. \tag{7.2.2}$$

先定义一个概念, 称为形式.

定义 7.2.2 给定 $t \geqslant 2$, 一个 t 形式是一组 t 个 $t \times t$ 逻辑函数, 即

$$T = \left\{T_1, T_2, \cdots, T_t \middle| T_i \in \mathcal{L}_{t\times t},\ 1 \leqslant i \leqslant t\right\}.$$

注 设 $F : \mathcal{D}_t \times \mathcal{D}_t \to \mathcal{D}_t$, 称 F 为一 t 门. 设其结构矩阵为

$$M_F = [M_1\ M_2\ \cdots\ M_t],$$

这里, $M_i \in \mathcal{L}_{t\times t}$, $i = 1, \cdots, t$. 那么, t 形式

$$T_F = \{M_1, M_2, \cdots, M_t\}$$

称为由 t 门 F 生成的 t 形式.

反之, 给定一个 t 形式, 则可将其元素排成一行构成一个 t 门的结构矩阵, 从而决定一个 t 门. 由于元素排列顺序的不同, 由 t 形式生成的 t 门不唯一 (最多可达 $t!$ 个). 后面会证明: M_i 在 M_F 中的排序与复合分解无关. 因此, 如果不加说明, 就假定如果 F 由形式 $T=\{T_i|i=1,\cdots,t\}$ 生成, 则 $M_F=[T_1,\cdots,T_t]$.

例 7.2.1 考察最大门与最小门, 分别记为 $\vee$ 和 $\wedge$.

- $t=2$: 我们有 $M_\vee=\delta_2[1\ 1\ 1\ 2]$ 及 $M_\wedge=\delta_2[1\ 2\ 2\ 2]$. 于是

$$
\begin{aligned}
T_\vee&=\{\delta_2[1\ 1],\delta_2[1\ 2]\};\\
T_\wedge&=\{\delta_2[1\ 2],\delta_2[2\ 2]\}.
\end{aligned}
$$

- $t=3$: 不难算出

$$
\begin{aligned}
M_\vee&=\delta_3[1\ 1\ 1\ 1\ 2\ 2\ 1\ 2\ 3];\\
M_\wedge&=\delta_3[1\ 2\ 3\ 2\ 2\ 3\ 3\ 3\ 3].
\end{aligned}
$$

因此,

$$
\begin{aligned}
T_\vee&=\{\delta_3[1\ 1\ 1],\delta_3[1\ 2\ 2],\delta_3[1\ 2\ 3]\};\\
T_\wedge&=\{\delta_3[1\ 2\ 3],\delta_3[2\ 2\ 3],\delta_3[3\ 3\ 3]\}.
\end{aligned}
$$

- $t=4$: 类似地可得

$$
\begin{aligned}
T_\vee&=\{\delta_4[1\ 1\ 1\ 1],\delta_4[1\ 2\ 2\ 2],\delta_4[1\ 2\ 3\ 3],\delta_4[1\ 2\ 3\ 4]\};\\
T_\wedge&=\{\delta_4[1\ 2\ 3\ 4],\delta_4[2\ 2\ 3\ 4],\delta_4[3\ 3\ 3\ 4],\delta_4[4\ 4\ 4\ 4]\}.
\end{aligned}
$$

现在考虑分割 (7.2.2), 令 $p=\prod_{i=1}^r k_i$, $q=\prod_{i=r+1}^n k_i$, $k=pq$. 记 $x^1=\ltimes_{i=1}^r x_i$, $x^2=\ltimes_{i=r+1}^n x_i$, 且多值逻辑函数 $f:\prod_{i=1}^n\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$ 的结构矩阵为

$$
M_f=[M_1\ M_2\ \cdots\ M_p], \tag{7.2.3}
$$

这里 $M_i\in\mathcal{L}_{k_0\times q}$, $i=1,\cdots,p$. 那么, 有以下定理.

定理 7.2.1 设多值逻辑函数 $f:\prod_{i=1}^n\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$ 的结构矩阵为 (7.2.3). 它具有不相交复合分解 (7.2.2), 当且仅当, 存在

(i) 一个 k_0 形式 $T=\{T_1,T_2,\cdots,T_{k_0}\}\subset\mathcal{L}_{k_0\times k_0}$, 以及

(ii) 一个逻辑矩阵 $M_\psi\in\mathcal{L}_{k_0\times q}$,

使得

$$
M_i=T_{s_i}M_\psi,\quad 其中\ T_{s_i}\in T,\ i=1,\cdots,p. \tag{7.2.4}
$$

证明 (必要性) 假定有三个函数 F, ϕ, 及 ψ 使复合分解 (7.2.1) 成立. 记 f 的结构矩阵为 $M_f\in\mathcal{L}_{k_0\times k}$, 并将其分划成 (7.2.3) 的形式. 设 F 的结构矩阵为

$$
M_F=[F_1\ F_2\ \cdots\ F_{k_0}],
$$

这里 $F_i \in \mathcal{L}_{k_0 \times k_0}$, $i=1,\cdots,k_0$; 又设 ϕ 的结构矩阵为

$$M_\phi = \delta_{k_0}[i_1\ i_2\ \cdots\ i_p];$$

以及 ψ 的结构矩阵为 $M_\psi \in \mathcal{L}_{k_0 \times q}$. 那么有

$$M_f x = M_F M_\phi x^1 M_\psi x^2,$$

这里 $x = \ltimes_{i=1}^n x_i$, $x^1 = \ltimes_{i=1}^r x_i$, $x^2 = \ltimes_{i=r+1}^n x_i$.

利用等式 (1.3.8) 可得

$$M_f = M_F M_\phi (I_p \otimes M_\psi). \tag{7.2.5}$$

首先计算 $M_F M_\phi$, 记其为

$$M_F M_\phi := [N_1\ N_2\ \cdots\ N_p].$$

直接计算可知

$$N_j = F_{i_j}, \quad j=1,\cdots,p.$$

如果记

$$M_F M_\phi (I_p \otimes M_\psi) := [W_1\ W_2\ \cdots\ W_p], \tag{7.2.6}$$

那么

$$W_j = F_{i_j} M_\psi, \quad j=1,\cdots,p. \tag{7.2.7}$$

分别用 (7.2.3) 以及 (7.2.6)–(7.2.7) 代入 (7.2.5) 的左边与右边, 然后比较两边相应的矩阵块可知

$$M_i = F_{i_j} M_\psi, \quad i=1,\cdots,p. \tag{7.2.8}$$

设

$$T = T_F = \{F_1, F_2, \cdots, F_{k_0}\},$$

则 (7.2.8) 证明了必要性.

(充分性) 假定存在一个 k_0 形式 $T=\{T_1,T_2,\cdots,T_{k_0}\}$ 以及一个 M_ψ 使得

$$M_i = T_{j_i} M_\psi, \quad i=1,\cdots,p. \tag{7.2.9}$$

令一个 k_0 门 F 由其结构矩阵确定, 即

$$M_F = [T_1\ T_2\ \cdots\ T_{k_0}],$$

并且

$$M_\phi = \delta_{k_0}[j_1\ j_2\ \cdots\ j_p].$$

那么, 直接计算可得

$$M_F M_\phi (I_p \otimes M_\psi) = M_f,$$

这表明

$$F(\phi(x^1), \psi(x^2)) = f(x).$$

□

将定理 7.2.1 用于布尔函数的情况, 则有如下推论:

推论 7.2.1 考察一个布尔函数 $f(x_1, \cdots, x_n)$. 它相对于分割 (7.2.2) 可复合分解, 当且仅当, 存在一个 2 形式 T 和一个逻辑矩阵 $M_\psi \in \mathcal{L}_{2\times 2^{n-r}}$, 使 f 的结构矩阵具有如下形式

$$M_f = [\mu_1 M_\psi\ \mu_2 M_\psi\ \cdots\ \mu_{2^r} M_\psi],$$

这里 $\mu_i \in T$, $\forall i$.

注 注意到

$$\mathcal{L}_{2\times 2} = \{\mathrm{T}_1 := \delta_2[1,1],\ \mathrm{T}_2 := \delta_2[1,2], \mathrm{T}_3 := \delta_2[2,1],\ \mathrm{T}_4 := \delta_2[2,2]\}.$$

不考虑元素顺序, 我们有 10 个可能的 2 形式 $T = \{\mathrm{T}_i, \mathrm{T}_j\}$, $1 \leqslant i, j \leqslant 4$, 其逻辑形式见表 7.2.1.

表 7.2.1 2 形式及其相应的逻辑函数

$\{\mathrm{T}_1,\mathrm{T}_1\}$	$\{\mathrm{T}_1,\mathrm{T}_2\}$	$\{\mathrm{T}_1,\mathrm{T}_3\}$	$\{\mathrm{T}_1,\mathrm{T}_4\}$	$\{\mathrm{T}_2,\mathrm{T}_2\}$
1	$\phi \vee \psi$	$\phi \vee \neg\psi$	ϕ	ψ
$\{\mathrm{T}_2,\mathrm{T}_3\}$	$\{\mathrm{T}_2,\mathrm{T}_4\}$	$\{\mathrm{T}_3,\mathrm{T}_3\}$	$\{\mathrm{T}_3,\mathrm{T}_4\}$	$\{\mathrm{T}_4,\mathrm{T}_4\}$
$\phi \leftrightarrow \psi$	$\phi \wedge \psi$	$\neg\psi$	$\neg(\phi \to \psi)$	0

略去 (i) 平凡情况: $\Gamma = \varnothing$ 或/且 $\Lambda = \varnothing$; (ii) 重复情况: $\phi \vee \psi$ 与 $\phi \vee \neg\psi$ 等价, 视为重复; 剩下的只有四种不同情况:

(i) 类型 1: $T^1 = \{\mathrm{T}_1, \mathrm{T}_2\}$, 即, $F(\phi, \psi) = \phi \vee \psi$;

(ii) 类型 2: $T^2 = \{\mathrm{T}_2, \mathrm{T}_3\}$ 即, $F(\phi, \psi) = \phi \leftrightarrow \psi$;

(iii) 类型 3: $T^3 = \{\mathrm{T}_2, \mathrm{T}_4\}$ 即, $F(\phi, \psi) = \phi \wedge \psi$;

(iv) 类型 4: $T^4 = \{\mathrm{T}_3, \mathrm{T}_4\}$ 即, $F(\phi, \psi) = \neg(\phi \to \psi)$.

注　理论上说, 因为定理 7.2.1 只涉及到有限集, 因此是可验证的. 但实际上, 当 $k_0 > 2$ 时, 计算复杂性可能很大. 我们给出以下程序以检验相关条件.

(i) 第一步: 将 M_f 等分为 p 块, 如 (7.2.3).

(ii) 第二步: 定义矩阵

$$\Psi := \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_p \end{bmatrix}. \tag{7.2.10}$$

检验是否满足

$$\operatorname{rank}(\Psi) \leqslant \mathrm{k}_0. \tag{7.2.11}$$

如果“否”, 那么, 对于这个 p 及其相应的分割, 复合分解不可行. 算法停止. (注意, 条件 (7.2.11) 来自 (7.2.4).)

如果“是”, 进入下一步.

(iii) 第三步: 如果存在 Ψ 的线性独立的行以构造一个 $k_0 \times q$ 维逻辑矩阵, 记其为 M_ψ. 进入下一步;

否则, 函数不可分解. (注意, 如果 $\operatorname{rank}(\Psi) < k_0$, 那么, 可以加上一些其他线性无关行以构成 Ψ. 这时我们有多于一个, 但依然是有限个的 M_ψ.)

(iv) 第四步: 利用这个 M_ψ, 解代数方程 (7.2.4) 以求出 $T_{s_i}, i = 1, \cdots, p$. (只有当 (a) 解存在; (b) $T_{s_i} \in \mathcal{L}_{k_0 \times k_0}$; 及 (c) 共有 $s \leqslant k_0$ 不同解, 即, $|\{T_{s_i} | i = 1, \cdots, p\}| \leqslant k_0$, 然后, 复合分解才存在.)

下面的例子用于描述定理 7.2.1 和以上注释.

例 7.2.2　考虑 3 值函数 $f(x_1, x_2, x_3, x_4) : \mathcal{D}_3^4 \to \mathcal{D}_3$, 设其结构矩阵为

$$\begin{aligned} M_f = \delta_3[&1\,1\,1\,1\,1\,1\,1\,1\,1\;\;1\,2\,2\,2\,2\,2\,2\,2\,1\;\;1\,2\,3\,2\,2\,2\,3\,2\,1 \\ &1\,1\,1\,1\,1\,1\,1\,1\,1\;\;1\,2\,2\,2\,2\,2\,2\,2\,1\;\;1\,2\,2\,2\,2\,2\,2\,2\,1 \\ &1\,1\,1\,1\,1\,1\,1\,1\,1\;\;1\,1\,1\,1\,1\,1\,1\,1\,1\;\;1\,1\,1\,1\,1\,1\,1\,1\,1]. \end{aligned}$$

考虑其相对于分割 $\Gamma = \{1,2\}$, $\Lambda = \{3,4\}$ 的复合分解, 则 $p = q = 9$. 记

$M_f = [M_1\ M_2\ \cdots\ M_9]$, 那么

$$\Psi = \begin{bmatrix} \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1] \\ \delta_3[1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1] \\ \delta_3[1\ 2\ 3\ 2\ 2\ 2\ 3\ 2\ 1] \\ \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1] \\ \delta_3[1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1] \\ \delta_3[1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1] \\ \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1] \\ \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1] \\ \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1] \end{bmatrix}.$$

容易检验 $\mathrm{rank}(\Psi) = k_0 = 3$. 我们可选 Ψ 的线性无关行以构成逻辑矩阵 M_ψ. 例如, 选 Ψ 的 7, 8, 9 行 (即, M_3), 则

$$M_\psi = \delta_3[1\ 2\ 3\ 2\ 2\ 2\ 3\ 2\ 1].$$

然后, 我们解 (7.2.4) 以求 T_1, T_2, T_3. 即

$$\begin{aligned} T_1 M_\psi &= \delta_3[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1], \\ T_2 M_\psi &= \delta_3[1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1], \\ T_3 M_\psi &= \delta_3[1\ 2\ 3\ 2\ 2\ 2\ 3\ 2\ 1]. \end{aligned}$$

一个有效的解为 $T = \{T_1 = \delta_3[1\ 1\ 1],\ T_2 = \delta_3[1\ 2\ 2],\ T_3 = \delta_3[1\ 2\ 3]\}$. 于是可选

$$M_F = \delta_3[1\ 1\ 1\ \ 1\ 2\ 2\ \ 1\ 2\ 3].$$

最后, 由 $M_f = [T_1, T_2, T_3, T_1, T_2, T_2, T_1, T_1, T_1] M_\psi$ 可得

$$M_\phi = \delta_3[1\ 2\ 3\ \ 1\ 2\ 2\ \ 1\ 1\ 1].$$

如果利用基本逻辑算子 $\{\vee, \wedge, \neg\}$ 定义 3 值逻辑算子 $\to$ 和 $\leftrightarrow$ 如下:

$$\begin{aligned} A \to B &:= \neg A \vee B, \\ A \leftrightarrow B &:= (A \to B) \wedge (B \to A), \end{aligned}$$

则可得到

$$M_{\to} = M_\phi, \quad M_{\leftrightarrow} = M_\psi.$$

因此, 有 f 的复合分解形式如下:

$$f(x_1, x_2, x_3, x_4) = (x_1 \to x_2) \vee (x_3 \leftrightarrow x_4).$$

命题 7.2.1　因子 ϕ 与 ψ 的先后顺序与可分解性无关.

证明　设

$$f(X) = F(\phi(X_1), \psi(X_2))$$

的代数形式为 $f(x) = M_f x$, 那么

$$M_f x = M_F M_\phi x^1 M_\psi x^2 = M_F W_{[k_0,k_0]} M_\psi x^2 M_\phi x^1.$$

因此, $M_f x = \tilde{F}(\phi(X_2), \psi(X_1))$, 这里 $\tilde{F}$ 的结构矩阵为 $M_{\tilde{F}} := M_F W_{[k_0,k_0]}$. 由于我们考虑所有可能的 M_F, 而 $M_{\tilde{F}} := M_F W_{[2,2]}$ 是其中一种 M_F, 并且对这种 $M_{\tilde{F}}$ 只是 ϕ 和 ψ 的前后顺序变了. □

注　1) 根据命题 7.2.1, 可以选择 $k \leqslant n/2$ 变量作为第二个函数 ψ 的自变量, 这就已穷尽所有可能.

2) 如果 (7.2.2) 不成立. 此时 $\{\Gamma, \Lambda\}$ 为 $\{1, 2, \cdots, n\}$ 的一个任意分割 (但 $\Gamma \neq \varnothing$ 且 $\Lambda \neq \varnothing$). 这时, 只要对变量位置做一个重新排列即可. 例如, 考虑 f 为布尔函数. 设 $\Lambda = \{j_1, j_2, \cdots, j_s\}$ 是一组变量, 这里 $s \leqslant n_0$ 且 $j_1 < j_2 < \cdots < j_s$, 而 $\Gamma = \Lambda^c$. 记

$$x^1 := \ltimes_{i\in\Gamma} x_i; \quad x^2 = \ltimes_{i=1}^s x_{j_i}.$$

则有

$$\begin{aligned}
f(x_1, \cdots, x_n) &= M_f \ltimes_{i=1}^n x_i \\
&= M_f W_{[2,2^{j_s-1}]} x_{j_s} \ltimes_{i\neq j_s} x_i \\
&= M_f W_{[2,2^{j_s-1}]} W_{[2,2^{j_{s}-1}]} x_{j_{s-1}} x_{j_s} \ltimes_{i\notin\{j_s, j_{s-1}\}} x_i \\
&= \cdots \\
&= M_f W_{[2,2^{j_s-1}]} W_{[2,2^{j_{s}-1}]} \cdots W_{[2,2^{j_1+(s-2)}]} \ltimes_{i=1}^s x_{j_i} \ltimes_{i\in\Gamma} x_i \\
&= M_f W_{[2,2^{j_s-1}]} W_{[2,2^{j_{s}-1}]} \cdots W_{[2,2^{j_1+(s-2)}]} W_{[2^{n-s},2^s]} x^1 x^2.
\end{aligned}$$

由上面的讨论可知, 设 $\{x_{j_1}, \cdots, x_{j_s}\}$ 被选为 ψ 的自变量, 那么, f 对应于 $x^1 = \ltimes_{i\in\Gamma} x_i$ 和 $x^2 = \ltimes_{i\in\Lambda} x_i$ 的结构矩阵变为

$$\tilde{M}_f = M_f \ltimes_{k=s}^1 W_{[2,2^{j_k+(s-1)-k}]} \ltimes W_{[2^{n-s},2^s]}. \tag{7.2.12}$$

7.3　相交复合分解

定义 7.3.1　设 $f: \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 为一多值映射, $\Gamma \cup \Theta \cup \Lambda$ 为 $\{1, 2, \cdots, n\}$ 的一个分割. f 称为对分割 $\Gamma \cup \Theta \cup \Lambda$ 的一个相交复合分解, 如果存在 k_0 值函数

$F:\mathcal{D}_{k_0}^2\to\mathcal{D}_{k_0}$, 和两个多值函数 $\phi:\prod_{i\in\Gamma\cup\Theta}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$ 和 $\psi:\prod_{i\in\Lambda\cup\Theta}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$, 使得

$$f(x_1,\cdots,x_n)=F\left(\phi(x_\gamma),\psi(x_\lambda)\right), \tag{7.3.1}$$

其中 $x_\gamma\in\prod_{i\in\Gamma\cup\Theta}\mathcal{D}_{k_i}$, $x_\lambda\in\prod_{i\in\Lambda\cup\Theta}\mathcal{D}_{k_i}$.

同样, 假定分割是良序的, 即存在 $r_1\geqslant 1$, $r_2\geqslant 1$ 且 $r_1+r_2<n$, 使得 $\Gamma=\{1,2,\cdots,r_1\},\Theta=\{r_1+1,r_1+2,\cdots,r_1+r_2\},\Lambda=\{r_1+r_2+1,r_1+r_2+2,\cdots,n\}$.

为叙述方便, 记

$$\prod_{i=1}^{r_1}k_i=p,\quad \prod_{i=r_1+1}^{r_1+r_2}k_i=\ell,\quad \prod_{i=r_1+r_2+1}^{n}k_i=q,$$

以及

$$x^1=\ltimes_{i=1}^{r_1}x_i,\quad x^2=\ltimes_{i=r_1+1}^{r_1+r_2}x_i,\quad x^3=\ltimes_{i=r_1+r_2+1}^{n}x_i.$$

设 f 的结构矩阵为 M_f, 将其等分为 ℓp 块如下

$$M_f=[M_1^1\ M_1^2\ \cdots\ M_1^\ell\ \ \cdots\ M_p^1\ M_p^2\ \cdots\ M_p^\ell]. \tag{7.3.2}$$

那么, 我们有如下定理:

定理 7.3.1 考察多值映射 $f:\prod_{i=1}^n\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$, 其结构矩阵为 M_f. f 相对于分割 Γ,Θ,Λ 具有复合分解 (7.3.1), 当且仅当, 存在 (i) 一个 k_0 形式 T, 和 (ii) 一组逻辑矩阵

$$M_\psi^s\in\mathcal{L}_{k_0\times q},\quad s=1,\cdots,\ell,$$

使得 f 的结构矩阵可表示为

$$\begin{aligned}M_f=[&\mu_{1,1}M_\psi^1\ \mu_{1,2}M_\psi^2\ \cdots\ \mu_{1,\ell}M_\psi^\ell\\ &\mu_{2,1}M_\psi^1\ \mu_{2,2}M_\psi^2\ \cdots\ \mu_{2,\ell}M_\psi^\ell\\ &\cdots\\ &\mu_{p,1}M_\psi^1\ \mu_{p,2}M_\psi^2\ \cdots\ \mu_{p,\ell}M_\psi^\ell,],\end{aligned} \tag{7.3.3}$$

这里

$$\mu_{i,j}\in T,\quad i=1,\cdots,p;\ j=1,\cdots,\ell.$$

证明 (必要性) 设存在 F, ϕ, ψ, 使得 (7.3.1) 成立. 并设 F 的结构矩阵表示为

$$M_F=[F_1\ F_2\ \cdots\ F_{k_0}],$$

这里 $F_i \in \mathcal{L}_{k_0 \times k_0}$, $i = 1, \cdots, k_0$; 又设 ϕ 的结构矩阵为

$$M_\phi = \delta_{k_0}[j_1\ j_2\ \cdots\ j_{p\ell}];$$

ψ 的结构矩阵为

$$M_\psi = \left[M_\psi^1\ M_\psi^2\ \cdots\ M_\psi^\ell\right] \in \mathcal{L}_{k_0 \times q\ell},$$

这里

$$M_\psi^i \in \mathcal{L}_{k_0 \times q}, \quad i = 1, \cdots, \ell.$$

那么, 可以构造一个 k_0 形式如下:

$$T := \{F_1, F_2, \cdots, F_{k_0}\}.$$

根据 (7.3.1), 有

$$M_f x = M_F M_\phi x^1 x^2 M_\psi x^2 x^3,$$

这里 $x = \ltimes_{i=1}^n x_i$.

利用等式 (1.3.8) 及 (3.1.3), 可得下式

$$M_f = M_F M_\phi \left(I_{p\ell} \otimes M_\psi\right)\left(I_p \otimes M_{r,\ell}\right). \tag{7.3.4}$$

我们先计算 $M_F M_\phi$, 它可记为

$$M_F M_\phi := [N_1\ N_2\ \cdots\ N_{p\ell}]. \tag{7.3.5}$$

类似不相交情况, 有

$$N_s \in T, \quad s = 1, 2, \cdots, p\ell.$$

下面计算 $\left(I_{p\ell} \otimes M_\psi\right)\left(I_p \otimes M_{r,\ell}\right)$:

$$\left(I_{p\ell} \otimes M_\psi\right)\left(I_p \otimes M_{r,\ell}\right) = I_p \otimes \left[\left(I_\ell \otimes M_\psi\right) M_{r,\ell}\right]. \tag{7.3.6}$$

首先简化 $\left(I_\ell \otimes M_\psi\right) M_{r,\ell}$. 注意到 $\left(I_\ell \otimes M_\psi\right) \in \mathcal{L}_{\ell k_0 \times \ell^2 q}$ 并且 $M_{r,\ell} \in \mathcal{L}_{\ell^2 \times \ell}$. 将它们转化为普通矩阵乘法形式, 则得

$$\left(I_\ell \otimes M_\psi\right) M_{r,\ell} = \left(I_\ell \otimes M_\psi\right)\left(M_{r,\ell} \otimes I_q\right). \tag{7.3.7}$$

而且, 有

$$I_\ell \otimes M_\psi = \left.\begin{bmatrix} M_\psi & 0 & \cdots & 0 \\ 0 & M_\psi & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & M_\psi \end{bmatrix}\right\} \ell; \tag{7.3.8}$$

以及

$$M_{r,\ell} \otimes I_q = \begin{bmatrix} \left.\begin{bmatrix} I_q \\ 0 \\ \vdots \\ 0 \end{bmatrix}\right\}\ell & 0 & \cdots & 0 \\ 0 & \left.\begin{bmatrix} 0 \\ I_q \\ \vdots \\ 0 \end{bmatrix}\right\}\ell & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & \left.\begin{bmatrix} 0 \\ 0 \\ \vdots \\ I_q \end{bmatrix}\right\}\ell \end{bmatrix}. \tag{7.3.9}$$

将 (7.3.8) 与 (7.3.9) 相乘可得

$$(I_\ell \otimes M_\psi)\, M_{r,\ell} = \begin{bmatrix} M_\psi^1 & 0 & \cdots & 0 \\ 0 & M_\psi^2 & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & M_\psi^\ell \end{bmatrix}. \tag{7.3.10}$$

将 (7.3.5)–(7.3.7) 以及 (7.3.10) 放到一起, 就可以得到 (7.3.4) 的右边. 而且, 不难发现 (7.3.4) 的右边具有 (7.3.3) 的样子. 这就证明了必要性.

(充分性) 设 k_0 形式

$$T = \{T_1, T_2, \cdots, T_{k_0}\}$$

给定, 利用它构造 k_0 门 F 如下

$$M_F = [T_1\ T_2\ \cdots\ T_{k_0}].$$

利用

$$M_\psi = \left[M_\psi^1\ M_\psi^2\ \cdots\ M_\psi^\ell\right]$$

为 ψ 的结构矩阵生成 ψ. 记

$$M_\phi = \left[M_\phi^{1,1}\ \cdots\ M_\phi^{1,\ell}\ \cdots\ M_\phi^{p,1}\ \cdots\ M_\phi^{p,\ell}\right].$$

由于 (7.3.3) 已知, 根据 $\mu_{\alpha,\beta}$, 可以依如下规则确定 $M_\phi^{\alpha,\beta}$: 如果 $\mu_{\alpha,\beta}=T_i$, 那么

$$M_\phi^{\alpha,\beta}=\delta_{k_0}^i. \tag{7.3.11}$$

由于结构矩阵可唯一确定其逻辑函数, $\{\phi,\psi\}$ 可由其上述结构矩阵得到. 最后, 由 (7.3.4) 不难验证分解式 (7.3.1) 成立. □

注 由定理 7.3.1 的证明不难看出, 当一个形式 T 确定, 其中元素 $\{T_i|i=1,\cdots,t\}$ 排成 M_F 时的顺序不影响可分解性. 因此, ϕ 与 ψ 的前后顺序同样不影响可分解性. 但是, 导出的 M_ϕ 可以不同.

下面给一个简单例子.

例 7.3.1 设 $f(x_1,x_2,x_3,x_4):\mathcal{D}_3^4\to\mathcal{D}_3$ 为一 3 值逻辑函数, 其结构矩阵为

$$\begin{aligned}M_f=\delta_3[&1\,1\,1\,2\,2\,2\,3\,3\,3\ \ 2\,2\,1\,2\,2\,2\,3\,3\,3\ \ 3\,2\,1\,3\,2\,2\,3\,3\,3\\ &1\,1\,1\,2\,2\,2\,3\,3\,3\ \ 2\,2\,2\,2\,2\,2\,2\,2\,2\ \ 2\,2\,2\,2\,2\,2\,2\,2\,2\\ &1\,1\,1\,2\,2\,2\,3\,3\,3\ \ 2\,2\,2\,2\,2\,2\,2\,2\,2\ \ 1\,2\,3\,1\,2\,2\,1\,1\,1].\end{aligned}$$

考虑相对于分割 $\Gamma=\{1\}$, $\Theta=\{2\}$, 及 $\Lambda=\{3,4\}$ 的分解. 于是, $p=\ell=3$, $q=9$. 将 M_f 分划为 $M_f=[M_1^1\ M_1^2\ M_1^3\ \ M_2^1\ M_2^2\ M_2^3\ \ M_3^1\ M_3^2\ M_3^3]$. 那么, 有

$$\Psi_1=\begin{bmatrix}\delta_3[1\,1\,1\,2\,2\,2\,3\,3\,3]\\ \delta_3[1\,1\,1\,2\,2\,2\,3\,3\,3]\\ \delta_3[1\,1\,1\,2\,2\,2\,3\,3\,3]\end{bmatrix};$$

$$\Psi_2=\begin{bmatrix}\delta_3[2\,2\,1\,2\,2\,2\,3\,3\,3]\\ \delta_3[2\,2\,2\,2\,2\,2\,2\,2\,2]\\ \delta_3[2\,2\,2\,2\,2\,2\,2\,2\,2]\end{bmatrix};$$

$$\Psi_3=\begin{bmatrix}\delta_3[3\,2\,1\,3\,2\,2\,3\,3\,3]\\ \delta_3[2\,2\,2\,2\,2\,2\,2\,2\,2]\\ \delta_3[1\,2\,3\,1\,2\,2\,1\,1\,1]\end{bmatrix}.$$

容易检验, $\mathrm{rank}(\Psi_i)=k_0=3, i=1,2,3$. 于是, 可以选择 Ψ_i 线性无关的行构成 M_ψ^i, $i=1,2,3$. 例如, 取

$$\begin{aligned}M_\psi^1&=\delta_3[1\,1\,1\,2\,2\,2\,3\,3\,3],\\ M_\psi^2&=\delta_3[2\,2\,1\,2\,2\,2\,3\,3\,3],\\ M_\psi^3&=\delta_3[3\,2\,1\,3\,2\,2\,3\,3\,3].\end{aligned}$$

于是

$$M_\psi = [M_\psi^1, M_\psi^2, M_\psi^3] = \delta_3[1\ 1\ 1\ 2\ 2\ 2\ 3\ 3\ 3\ 2\ 2\ 1\ 2\ 2\ 2\ 3\ 3\ 3\ \ 3\ 2\ 1\ 3\ 2\ 2\ 3\ 3\ 3].$$

下一步, 解如下方程

$$\begin{aligned}
\mu_{i,1}M_\psi^1 &= \delta_3[1\ 1\ 1\ 2\ 2\ 2\ 3\ 3\ 3], \quad i=1,2,3;\\
\mu_{1,2}M_\psi^2 &= \delta_3[2\ 2\ 1\ 2\ 2\ 2\ 3\ 3\ 3],\\
\mu_{i,2}M_\psi^2 &= \delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2], \quad i=2,3;\\
\mu_{1,3}M_\psi^3 &= \delta_3[3\ 2\ 1\ 3\ 2\ 2\ 3\ 3\ 3],\\
\mu_{2,3}M_\psi^3 &= \delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2],\\
\mu_{3,3}M_\psi^3 &= \delta_3[1\ 2\ 3\ 1\ 2\ 2\ 1\ 1\ 1].
\end{aligned}$$

一组有效解为

$$\begin{aligned}
&\mu_{1,1} = \mu_{2,1} = \mu_{3,1} = \mu_{1,2} = \mu_{1,3} = \delta_3[1\ 2\ 3];\\
&\mu_{2,2} = \mu_{2,3} = \mu_{3,2} = \delta_3[2\ 2\ 2];\\
&\mu_{3,3} = \delta_3[3\ 2\ 1].
\end{aligned}$$

因此, 可以选 $T = \{T_1 = \delta_3[1\ 2\ 3],\ T_2 = \delta_3[2\ 2\ 2],\ T_3 = \delta_3[3\ 2\ 1]\}$, 并构造

$$M_F = \delta_3[1\ 2\ 3\ \ 3\ 1\ 2\ \ 2\ 2\ 3].$$

然后, 根据 (7.3.11) 有

$$M_\phi = \delta_3[1\ 1\ 1\ 1\ 2\ 2\ 1\ 2\ 3].$$

由此, $f(x_1,x_2,x_3,x_4) = M_f x = M_F M_\phi x_1 x_2 M_\psi x_2 x_3 x_4$, 这里 M_F, M_ϕ, M_ψ 均见前.

如果返回到逻辑形式, 则不难得到

$$f(x_1,x_2,x_3,x_4) = (x_1 \vee x_2) \leftrightarrow [(x_2 \vee \neg x_4) \wedge x_3].$$

7.4 隐函数存在定理

作为复合分解的应用, 本节讨论逻辑函数的隐函数存在定理.

给定一组 r 个 k 值逻辑方程 ($r < n$):

$$g_j(x_1,\cdots,x_n) = 1, \quad j = 1,\cdots,r, \tag{7.4.1}$$

这里, $x_i \in \mathcal{D}_k$, $i=1,\cdots,n$, $g_j : \mathcal{D}_k^n \to \mathcal{D}_k$ 为 k 值逻辑函数. 注意, (7.4.1) 的右边可以是任意常数 $c_j \in \mathcal{D}_k$, 但为方便计, 选 $c_j = 1 \sim \delta_k^1$. 将一般形式变为这种形式是很简单的[60].

向量形式下有

$$x^1 = \ltimes_{i=1}^{n-r} x_i \in \Delta_p, \qquad 其中 \quad p = k^{n-r},$$
$$x^2 = \ltimes_{i=n-r+1}^{n} x_i \in \Delta_q, \qquad 其中 \quad q = k^{r}.$$

记 $g = (g_1, \cdots, g_r)$, 那么 $g:\ \mathcal{D}_k^n \to \mathcal{D}_k^r$ 是一个逻辑映射. 在向量形式下, 也可看作 $G:\ \Delta_p \times \Delta_q \to \Delta_q$. 不难将 (7.4.1) 表示为其代数形式:

$$M_g x^1 x^2 = \delta_q^1, \tag{7.4.2}$$

这里 M_g 是 g 的结构矩阵.

本节考虑, 什么时候 x^2 可以由 x^1 表出. 严格地说, (7.4.1) 可以表示为

$$x_j = \phi_j(x_1, \cdots, x_{n-r}), \quad j = n-r+1, \cdots, n, \tag{7.4.3}$$

这里 $\phi_j : \mathcal{D}_k^{n-r} \to \mathcal{D}_k$ 是适当的 k 值逻辑函数. 令 $\phi = (\phi_{n-r+1}, \cdots, \phi_n)$, 则 $\phi:\ \mathcal{D}_k^{n-r} \to \mathcal{D}_k^r$ 为一逻辑映射. 向量形式下, 它变为 $\phi:\ \Delta_p \to \Delta_q$.

这个问题的答案称为 k 值逻辑函数的隐函数存在定理. 我们可将其转化为特殊的多值逻辑复合分解问题. 设 $x, y \in \mathcal{D}_q$, 定义一个 q 门, 记作 "$\leftrightarrow$", 如下:

$$x \leftrightarrow y = \begin{cases} 1, & 如果\ x = y, \\ \xi, & 如果\ x \neq y, \end{cases} \tag{7.4.4}$$

这里 $\xi \in \mathcal{D}_q$ 为一定常值, 但 $\xi \neq 1$. 那么, (7.4.3) 可表示为

$$F_{\leftrightarrow}(\phi(x^1), x^2) = 1, \tag{7.4.5}$$

显然, 要让 (7.4.1) 和 (7.4.3) 等价, 其充要条件为

$$g(x^1, x^2) = F_{\leftrightarrow}(\phi(x^1), x^2). \tag{7.4.6}$$

这是我们利用复合分解导出全局隐函数存在定理的出发点.

注意, (7.4.4) 中的 $\xi \in \mathcal{D}_q$ 有多种选择, 因此, 满足 (7.4.6) 的逻辑算子 $\leftrightarrow$ 有许多. 这样, 我们就必须考虑所有能实现算子 $\leftrightarrow$ 的形式.

对于一个整数 $q > 1$ 定义一组矩阵, Ξ_i, 如下:

$$\Xi_i = \left\{ E_i \in \mathcal{L}_{q\times q} \middle| \operatorname{Col}_i(E_i) = \delta_q^1;\ \operatorname{Col}_j(E_i) \neq \delta_q^1,\ j \neq i \right\}, \tag{7.4.7}$$

这里, $i = 1, 2, \cdots, q$. 利用 Ξ_i, 构造一组 q 形式如下:

$$\mathcal{E}_q := \left\{ \{E_1, E_2, \cdots, E_q\} \middle| E_i \in \Xi_i,\ i = 1, 2, \cdots, q \right\}. \tag{7.4.8}$$

如前所述, 每一个 q 形式 $T \in \mathcal{E}_q$ 对应一个 q 门 $F : \mathcal{D}_q \times \mathcal{D}_q \to \mathcal{D}_q$, 它以

$$E_T := [E_1\ E_2\ \cdots\ E_q]$$

为其结构矩阵, 即 $M_f = E_T$.

下面的引理表明, $\mathcal{E}$ 是对应于算子 $\leftrightarrow$ 的形式集合.

引理 7.4.1 设 $x, y \in \Delta_q$. 那么 $x = y$, 当且仅当, 存在 $T \in \mathcal{E}_q$ 使得

$$E_T xy = \delta_q^1. \tag{7.4.9}$$

证明 记

$$E_T = [E_1\ E_2\ \cdots\ E_q],$$

并令 $x = \delta_q^\alpha$, $y = \delta_q^\beta$. 直接计算可得

$$E_T xy = \mathrm{Col}_\beta(E_\alpha).$$

因此, (7.4.9) 成立, 当且仅当, $\alpha = \beta$. □

由此可得如下定理:

定理 7.4.1(隐函数存在定理) 记 g 的结构矩阵为 $M_g = [M_1\ M_2, \cdots, M_q]$. 那么, x_j, $j = n - r + 1, \cdots, n$ 可由 (7.4.1) 解出, 即表示为 (7.4.3) , 当且仅当, 存在一个 q 形式

$$T = \{E_1, E_2, \cdots, E_q\} \in \mathcal{E}_q,$$

使得

$$M_i \in T, \quad i = 1, \cdots, q. \tag{7.4.10}$$

证明 将 (7.4.3) 表示为其代数形式

$$x^2 = M_\phi x^1.$$

由引理 7.4.1, (7.4.1) 能表示为 (7.4.3), 当且仅当, 存在一个 $T \in \mathcal{E}_q$, 使得其对应的 q 门 F, 即 $M_F = E_T$, 满足

$$M_F M_\phi x^1 x^2 = \delta_q^1. \tag{7.4.11}$$

比较等式 (7.4.11) 与 (7.4.2), 显然, 其充要条件是 g 可表示成

$$M_g x^1 x^2 = M_F M_\phi x^1 x^2, \tag{7.4.12}$$

这里 M_F 对应于 $T \in \mathcal{E}_q$.

现在, 形式地考虑 $x^2 = M_\psi x^2$, 这里 $M_\psi = I_q$, 由引理 7.4.1 及定理 7.2.1 即得结论. □

7.5　注释与参考

逻辑函数的复合分解在线路板设计中可以减少线路的面积、通讯时延和耗电, 因此, 有很大实用价值[83]. 分解是一个比较古老的问题, 不但对布尔函数, 而且对一般逻辑函数也有许多讨论, 一些经典的结果可见文献 [67, 111, 120, 128-130]. 有一种算法称为 Quine-McCluskey 方法, 它对小型线路很有效[71]. 对于大型线路的算法探讨可见文献 [20, 68, 127].

布尔函数的隐函数存在定理曾在文献 [22] 中讨论过. 它提供了局部的充分条件. 定理 7.4.1 是对 k 值逻辑函数给出全局的充要条件, 应当是对已知结果的很大改进. 对更一般的多值逻辑函数目前尚无结果.

本章内容主要基于文献 [62].

第 8 章　布尔函数的微积分

8.1 布尔导数

布尔导数有多种不同定义, 本章采用的布尔导数来自文献 [28], 它是最普遍采用的一种.

定义 8.1.1　设 $f(x_1,\cdots,x_n):\mathcal{D}^n\to\mathcal{D}$ 为一布尔函数.

1) f 对变量 x_i 的布尔导数定义为

$$\frac{\partial f}{\partial x_i}=f(x_1,\cdots,x_i,\cdots,x_n)\langle+\rangle f(x_1,\cdots,\neg x_i,\cdots,x_n). \tag{8.1.1}$$

2) f 对 $x_{i_1},\cdots,x_{i_k}$ 的高阶布尔导数定义为

$$\frac{\partial^k f}{\partial x_{i_1}\cdots\partial x_{i_k}}=\frac{\partial}{\partial x_{i_1}}\left(\frac{\partial}{\partial x_{i_2}}\left(\cdots\left(\frac{\partial f}{\partial x_{i_k}}\right)\right)\right). \tag{8.1.2}$$

注意: 在本章 $\langle+\rangle:=\bar{\vee}$, 即, $+\ (\text{mod } 2)$, $\langle\ltimes\rangle:=\ltimes\ (\text{mod } 2)$.

下面列举一些由定义可直接导出的性质:

命题 8.1.1 ([141])　1) 设 $c\in\mathcal{D}$ 为一常数, 则

$$\frac{\partial c}{\partial x_i}=0. \tag{8.1.3}$$

2) $\dfrac{\partial f}{\partial x_i}$ 与 x_i 无关, 因此

$$\frac{\partial^2 f}{\partial^2 x_i}=0. \tag{8.1.4}$$

3)

$$\frac{\partial x_i}{\partial x_i}=1. \tag{8.1.5}$$

4)

$$\frac{\partial^2 f}{\partial x_i\partial x_j}=\frac{\partial^2 f}{\partial x_j\partial x_i}. \tag{8.1.6}$$

5)

$$\frac{\partial(f_1 \langle + \rangle f_2)}{\partial x_i} = \frac{\partial f_1}{\partial x_i} \langle + \rangle \frac{\partial f_2}{\partial x_i}. \tag{8.1.7}$$

6)

$$\frac{\partial(f_1 f_2)}{\partial x_i} = \frac{\partial f_1}{\partial x_i} f_2 \langle + \rangle f_1 \frac{\partial f_2}{\partial x_i} \langle + \rangle \frac{\partial f_1}{\partial x_i}\frac{\partial f_2}{\partial x_i}. \tag{8.1.8}$$

7)

$$\frac{\partial(x_i x_j)}{\partial x_i} = x_j. \tag{8.1.9}$$

8) 记 $\bar{f} := \neg f$, 及 $\bar{x} := \neg x$, 那么

$$\frac{\partial \bar{f}}{\partial x_i} = \frac{\partial f}{\partial x_i}, \quad \text{并且} \quad \frac{\partial f}{\partial \bar{x}_i} = \frac{\partial f}{\partial x_i}. \tag{8.1.10}$$

我们的第一个目标是导数 $\dfrac{\partial}{\partial x_i}$ 的结构矩阵, 记作 $M_{\partial_i f}$. 设 f 的结构矩阵为 M_f, $x := \ltimes_{i=1}^n x_i$. 利用 (8.1.1) 可得

$$\frac{\partial f}{\partial x_i} = M_{\partial_i f} x = M_f x \langle + \rangle M_f x_1 \cdots \neg x_i \cdots x_n. \tag{8.1.11}$$

利用向量形式, 一个简单代数变换即可将 (8.1.11) 的右边变为标准形式, 这个就是 $\dfrac{\partial}{\partial x_i}$ 的结构矩阵[104]

$$M_{\partial_i f} = M_{\bar{\vee}} M_f \left[I_{2^n} \otimes M_f \left(I_{2^{i-1}} \otimes M_n\right)\right] M_{r,2^n}, \quad 1 \leqslant i \leqslant n. \tag{8.1.12}$$

至于高阶导数, 也可以逐次算得.

下面给出导数结构矩阵的准确形式. 定义 $g(x_1, \cdots, x_n) := f(x_1, \cdots, \neg x_i, \cdots, x_n)$. 设 f 与 g 的结构矩阵分别为 M_f 和 M_g . 直接计算可知

$$\begin{aligned} M_g x &= M_f x_1 \cdots x_{i-1} M_n x_i \cdots x_n \\ &= M_f \left(I_{2^{i-1}} \otimes M_n\right) x, \end{aligned}$$

即

$$M_g = M_f \left(I_{2^{i-1}} \otimes M_n\right). \tag{8.1.13}$$

记 $m_f = \mathrm{Row}_1(M_f)$ 及 $m_g = \mathrm{Row}_1(M_g)$. 实际上, $m_f = T_f^{\mathrm{T}}$, 这里 T_f 是真值向量. 为方便, 也称 m_f 为 f 的真值向量.

下面的性质是显见的.

命题 8.1.2 设 $f(x_1,\cdots,x_n)$ 和 $g(x_1,\cdots,x_n)$ 的真值向量分别为 $m_f,m_g\in\mathcal{B}_{2^n}$, 而 σ 为一个二元逻辑算子. 那么

$$m_{f\sigma g}=m_f\sigma m_g. \tag{8.1.14}$$

注意, $m_f\sigma m_g=((m_f)_1\sigma(m_g)_1,\cdots,(m_f)_{2^n}\sigma(m_g)_{2^n})$.

利用命题 8.1.2, 有

$$m_{\partial_i f}=m_f\left\langle+\right\rangle m_f(I_{2^{i-1}}\otimes M_n). \tag{8.1.15}$$

利用矩阵半张量积的分配律可知

$$\begin{aligned}
m_{\partial_i f}&=m_f\left\langle+\right\rangle m_f(I_{2^{i-1}}\otimes M_n)\\
&=m_f\left\langle\ltimes\right\rangle I_{2^i}\left\langle+\right\rangle m_f\left\langle\ltimes\right\rangle(I_{2^{i-1}}\otimes M_n)\\
&=m_f\left\langle\ltimes\right\rangle(I_{2^i}\left\langle+\right\rangle(I_{2^{i-1}}\otimes M_n))\\
&=m_f\left\langle\ltimes\right\rangle(I_{2^{i-1}}\otimes(I_2\left\langle+\right\rangle M_n))\\
&=m_f\left\langle\ltimes\right\rangle(I_{2^{i-1}}\otimes\mathbf{1}_{2\times 2}).
\end{aligned}$$

由以上讨论可知

定理 8.1.1 设 $f(x_1,\cdots,x_n)$ 为一布尔函数且其结构矩阵为 M_f. 那么, 导数 $\dfrac{\partial f}{\partial x_i}$ 的矩阵 $M_{\partial_i f}$ 应为

$$M_{\partial_i f}=\begin{bmatrix}m_f\left\langle\ltimes\right\rangle\varXi_n^i\\ \neg m_f\left\langle\ltimes\right\rangle\varXi_n^i\end{bmatrix}, \tag{8.1.16}$$

这里

$$\varXi_n^i=I_{2^{i-1}}\otimes\mathbf{1}_{2\times 2}.$$

因此, 在向量形式下有

$$\frac{\partial f}{\partial x_i}=M_{\partial_i f}x, \tag{8.1.17}$$

这里, $x=\ltimes_{i=1}^n x_i$. 并且

$$m_{\partial_i f}=m_f\varXi_n^i. \tag{8.1.18}$$

由于我们知道 $\dfrac{\partial f}{\partial x_i}$ 与 x_i 无关, 故我们对以下形式有兴趣:

$$\frac{\partial f}{\partial x_i}=M_{\partial_{[i]}f}x_1\cdots x_{i-1}\hat{x}_i x_{i+1}\cdots x_n, \tag{8.1.19}$$

这里, 记号 "$\hat{x}_i$" 表示缺失 x_i.

为了计算 $M_{\partial_{[i]}f}$, 将 $M_{\partial_i f}$ 分裂为 2^i 个相同尺寸的块如下:

$$M_{\partial_i f} = [C_1 \, C_2 \, \cdots \, C_{2^i}].$$

不难发现, 要从 $M_{\partial_i f}$ 中找到 $M_{\partial_{[i]}f}$, 只须从前者中找出奇数 (或偶数) 块来. 它相当于右乘

$$\left(I_{2^{i-1}} \otimes \begin{bmatrix} I_{2^{n-i}} \\ \mathbf{0}_{2^{n-i}\times 2^{n-i}} \end{bmatrix} \right),$$

即

$$M_{\partial_{[i]}f} = \begin{bmatrix} m_f \,\langle\ltimes\rangle\, [\Psi_n^i]^{\mathrm{T}} \\ \neg m_f \,\langle\ltimes\rangle\, [\Psi_n^i]^{\mathrm{T}} \end{bmatrix}, \tag{8.1.20}$$

这里

$$\begin{aligned} \Psi_n^i &= \left(I_{2^{i-1}} \otimes \begin{bmatrix} I_{2^{n-i}} & \mathbf{0}_{2^{n-i}\times 2^{n-i}} \end{bmatrix} \right) \langle\ltimes\rangle \left(I_{2^{i-1}} \otimes \mathbf{1}_{2\times 2} \right) \\ &= I_{2^{i-1}} \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_{2^{n-i}}. \end{aligned}$$

注意到 M_f 的第一行 $\mathrm{Row}_1(M_f) = m_f$ 转置即为 f 的真值向量, 于是有

推论 8.1.1 设布尔函数 $f(x_1, \cdots, x_n)$ 的真值向量为 m_f. 那么, $\dfrac{\partial f}{\partial x_i}$ 的真值向量为

$$m_{\partial_{[i]}f}^{\mathrm{T}} = \Psi_n^i m_f^{\mathrm{T}}. \tag{8.1.21}$$

容易检验, 推论 8.1.1 与文献 [15, 16] 中的结论一致.

不难验证以下的推论 8.1.2 和 8.1.3, 它们在数值计算时很有用.

推论 8.1.2 将 m_f 等分为 2^i 块

$$m_f = \left[c_{1,1},\ c_{1,2},\ c_{2,1},\ c_{2,2},\ \cdots,\ c_{2^{i-1},1},\ c_{2^{i-1},2} \right].$$

那么, $m_{\partial_{[i]}f}$ 可直接由下式得到

$$m_{\partial_{[i]}f} = \left[c_{1,1} \langle+\rangle c_{1,2},\ c_{2,1} \langle+\rangle c_{2,2},\ \cdots,\ c_{2^{i-1},1} \langle+\rangle c_{2^{i-1},2} \right]. \tag{8.1.22}$$

推论 8.1.3 $\dfrac{\partial^k f}{\partial x_{i_1} \cdots \partial x_{i_k}}$ 的真值表在向量形式下为 (设 $i_1 > i_2 > \cdots > i_k$):

$$m_{\partial_{[i_1,\cdots,i_k]}f}^{\mathrm{T}} = \Psi_{n-k+1}^{i_k} \Psi_{n-k+2}^{i_{k-1}} \cdots \Psi_n^{i_1} m_f^{\mathrm{T}}. \tag{8.1.23}$$

注意: 在 (8.1.23) 中要求 $i_1 > i_2 > \cdots > i_k$, 否则后者的位置需要被调整. 例如, 设, $i_1 < i_2$, 那么, (参考后面的积分) 当对 $\mathrm{d}[i_1]$ 积分后, i_2 的位置会变为 $i_2 - 1$. 因此, 前述顺序是必要的. 再由 (8.1.6) 可知, 这个假设不失一般性.

下面讨论布尔函数的 MacLaurin 展式.

定理 8.1.2 ([17]) 一个布尔函数 $f(x_1, \cdots, x_n)$ 具有它的 MacLaurin 展式如下:

$$\begin{aligned} f(x_1,\cdots,x_n) =& f(\mathbf{0})\,\langle+\rangle \bigoplus_{i=1}^{n} \left.\frac{\partial f}{\partial x_i}\right|_{\mathbf{0}} \wedge x_i \,\langle+\rangle \bigoplus_{1\leqslant i_1<i_2\leqslant n} \left.\frac{\partial^2 f}{\partial x_{i_1}\partial x_{i_2}}\right|_{\mathbf{0}} \wedge x_{i_1} \wedge x_{i_2} \\ & \langle+\rangle \bigoplus_{1\leqslant i_1<i_2<i_3\leqslant n} \left.\frac{\partial^3 f}{\partial x_{i_1}\partial x_{i_2}\partial x_{i_3}}\right|_{\mathbf{0}} \wedge x_{i_1} \wedge x_{i_2} \wedge x_{i_3} \\ & \langle+\rangle \cdots \langle+\rangle \left.\frac{\partial^n f}{\partial x_1 \partial x_2 \cdots \partial x_n}\right|_{\mathbf{0}} \wedge x_1 \wedge x_2 \wedge \cdots \wedge x_n. \end{aligned} \tag{8.1.24}$$

给出一个例子如下:

例 8.1.1 设 $f(x_1, x_2, x_3, x_4) = (x_1 \bar{\vee} x_2) \to (x_3 \vee x_4)$. 它的真值表可计算如下:

$$m_f = [1\,1\,1\,1\,1\,1\,1\,0\,1\,1\,1\,0\,1\,1\,1\,1].$$

利用 (8.1.21),

$$\Psi_4^1 = \begin{bmatrix} I_8 & I_8 \end{bmatrix},$$

于是, $\dfrac{\partial f}{\partial x_1}$ 的真值表为

$$m_{\partial_{[1]} f} = m_f \,\langle\ltimes\rangle \left[\Psi_4^1\right]^{\mathrm{T}} = [0\,0\,0\,1\,0\,0\,0\,1].$$

类似地, 有

$$\Psi_4^2 = \begin{bmatrix} I_4 & I_4 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_4 & I_4 \end{bmatrix},$$

$$m_{\partial_{[2]} f} = m_f \,\langle\ltimes\rangle \left[\Psi_4^2\right] = [0\,0\,0\,1\,0\,0\,0\,1],$$

$$\Psi_4^3 = \begin{bmatrix} I_2 & I_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_2 & I_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & I_2 & I_2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & I_2 & I_2 \end{bmatrix},$$

$$m_{\partial_{[3]} f} = m_f \,\langle\ltimes\rangle \left[\Psi_4^3\right] = [0\,0\,0\,1\,0\,1\,0\,0],$$

$$\Psi_4^4 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & \vdots & & & & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix},$$

$$m_{\partial_{[4]}f} = m_f \langle \ltimes \rangle \left[\Psi_4^4\right] = [0\,0\,0\,1\,0\,1\,0\,0].$$

利用 (8.1.23), 可轻易算出

$$\begin{aligned} & m_{\partial_{[1,2]}f} = [0\,0\,0\,0]; \quad m_{\partial_{[1,3]}f} = [0\,1\,0\,1]; \quad m_{\partial_{[1,4]}f} = [0\,1\,0\,1]; \\ & m_{\partial_{[2,3]}f} = [0\,1\,0\,1]; \quad m_{\partial_{[2,4]}f} = [0\,1\,0\,1]; \quad m_{\partial_{[3,4]}f} = [0\,1\,1\,0]; \\ & m_{\partial_{[1,2,3]}f} = [0\,0]; \quad m_{\partial_{[1,2,4]}f} = [0\,0]; \quad m_{\partial_{[1,3,4]}f} = [1\,1]; \\ & m_{\partial_{[2,3,4]}f} = [1\,1]; \quad m_{\partial_{[1,2,3,4]}f} = 0. \end{aligned}$$

注意到计算布尔函数在 **0** 点的值, 就是要找到真值表最后一个数. 因此可得

$$\begin{aligned} & f(\mathbf{0}) = 1; && \frac{\partial f}{\partial x_1}|_{\mathbf{0}} = 1, && \frac{\partial f}{\partial x_2}|_{\mathbf{0}} = 1, && \frac{\partial f}{\partial x_3}|_{\mathbf{0}} = 0 \\ & \frac{\partial f}{\partial x_4}|_{\mathbf{0}} = 0; && \frac{\partial^2 f}{\partial x_1 \partial x_2}|_{\mathbf{0}} = 0, && \frac{\partial^2 f}{\partial x_1 \partial x_3}|_{\mathbf{0}} = 1, && \frac{\partial^2 f}{\partial x_1 \partial x_4}|_{\mathbf{0}} = 1, \\ & \frac{\partial^2 f}{\partial x_2 \partial x_3}|_{\mathbf{0}} = 1, && \frac{\partial^2 f}{\partial x_2 \partial x_4}|_{\mathbf{0}} = 1, && \frac{\partial^2 f}{\partial x_3 \partial x_4}|_{\mathbf{0}} = 0; && \frac{\partial^3 f}{\partial x_1 \partial x_2 \partial x_3}|_{\mathbf{0}} = 0, \\ & \frac{\partial^3 f}{\partial x_1 \partial x_2 \partial x_4}|_{\mathbf{0}} = 0, && \frac{\partial^3 f}{\partial x_1 \partial x_3 \partial x_4}|_{\mathbf{0}} = 1, && \frac{\partial^3 f}{\partial x_2 \partial x_3 \partial x_4}|_{\mathbf{0}} = 1; && \frac{\partial^4 f}{\partial x_1 \partial x_2 \partial x_3 \partial x_4}|_{\mathbf{0}} = 0. \end{aligned}$$

于是, 可得 $f(x)$ 的 MacLaurin 展式如下

$$\begin{aligned} f(x_1, x_2, x_3, x_4) = {} & 1 \langle + \rangle x_1 \langle + \rangle x_2 \langle + \rangle x_1 \wedge x_3 \langle + \rangle x_1 \wedge x_4 \langle + \rangle x_2 \wedge x_3 \\ & \langle + \rangle x_2 \wedge x_4 \langle + \rangle x_1 \wedge x_3 \wedge x_4 \langle + \rangle x_2 \wedge x_3 \wedge x_4. \end{aligned} \tag{8.1.25}$$

8.2 布尔代数方程与布尔微分方程

首先, 考虑带有已知布尔函数及其布尔导数的布尔等式的解.

定义 8.2.1 考察如下系统

$$\begin{cases} G_1\left(x_i, f, \dfrac{\partial f}{\partial x_i}, \cdots, \dfrac{\partial^k f}{\partial x_{i_1} \cdots \partial x_{i_k}}\right) = c_1, \\ G_2\left(x_i, f, \dfrac{\partial f}{\partial x_i}, \cdots, \dfrac{\partial^k f}{\partial x_{i_1} \cdots \partial x_{i_k}}\right) = c_2, \\ \quad \vdots \\ G_s\left(x_i, f, \frac{\partial f}{\partial x_i}, \cdots, \dfrac{\partial^k f}{\partial x_{i_1} \cdots \partial x_{i_k}}\right) = c_s. \end{cases} \tag{8.2.1}$$

1) 当 f 已知时 (8.2.1) 称为一个 (带微分的) 布尔代数方程, 其解 x 为满足方程的逻辑变量.

2) 当 f 未知时 (8.2.1) 称为 f 的一个布尔微分方程. 如果一个布尔函数 $g(x_1,\cdots,x_n)$ 满足 (8.2.1), 则称其为布尔微分方程的一个解.

3) 一组 $f(\mathbf{0})$ 及 f 在 $\mathbf{0}$ 点的布尔导数值称为布尔微分方程的初值, 一个布尔微分方程的解, 如果同时满足初值要求, 则称满足初值条件的解.

利用 (8.1.23), 解布尔代数方程 (8.2.1) 成为一个标准程序, 给出以下算法:

算法 8.2.1 • 第 1 步: 将逻辑方程转换为代数形式, 即

$$M_i x = c_i, \quad i=1,\cdots,s, \tag{8.2.2}$$

这里 $M_i \in \mathcal{L}_{2\times 2^n}$.

• 第 2 步: 将 (8.2.2) 各方程相乘以得到整体代数方程

$$Mx = c, \tag{8.2.3}$$

这里 $x=\ltimes_{i=1}^n x_i$, $c=\ltimes_{i=1}^s c_i$, 而 $M\in\mathcal{L}_{2^s\times 2^n}$ 由下式得到:

$$M = M_1 * M_2 * \cdots * M_s, \tag{8.2.4}$$

这里 $*$ 为 Khatri-Rao 积, 即

$$\mathrm{Col}_i(M) = \ltimes_{j=1}^s \mathrm{Col}_i(M_j), \quad i=1,\cdots,2^n.$$

• 第 3 步: 找出所有的解 $\delta_{2^n}^j$, 满足 $\mathrm{Col}_j(M)=c$.

这类问题的一个典型应用是组合逻辑电路的故障检测[90, 104]. 设 $f(x_1,\cdots,x_n)$ 为描述组合逻辑电路的布尔函数. 对于双固定故障的检测向量集 $x_i(s-a-\alpha)$, $x_j(s-a-\beta)$ 是下列方程的解集:

$$\bar{x}_i^\alpha x_j^\beta \frac{\partial f}{\partial x_i} \langle + \rangle x_i^\alpha \bar{x}_j^\beta \frac{\partial f}{\partial x_j} \langle + \rangle \bar{x}_i^\alpha \bar{x}_j^\beta \frac{\partial^2 f}{\partial x_i \partial x_j} = 1, \tag{8.2.5}$$

这里 $\alpha,\beta\in\mathcal{D}$, 并且 $x^1 := x$, $x^0 := \bar{x}$.

我们用一个例子来说明.

例 8.2.1 设一个组合逻辑电路描述如下[104]

$$\begin{aligned} f(x_1,\cdots,x_5) = &\neg\{\neg[x_2\vee(\neg x_1\wedge\neg x_3)]\}\vee\neg(x_1\vee x_5)\\ &\vee\neg(x_4\vee x_5)\vee\neg[\neg x_3\vee(\neg x_2\wedge\neg x_4)]. \end{aligned} \tag{8.2.6}$$

我们寻找双固定于 $x_3(s-a-1)$ 和 $x_4(s-a-0)$ 的检测向量集.

这个问题等价于解如下布尔代数方程

$$\bar{x}_3\bar{x}_4\frac{\partial f}{\partial x_3}\langle+\rangle x_3x_4\frac{\partial f}{\partial x_4}\langle+\rangle \bar{x}_3x_4\frac{\partial^2 f}{\partial x_3\partial x_4}=1, \tag{8.2.7}$$

其中 f 的结构矩阵为

$$M_f=\delta_2[1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 2\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 2].$$

利用推论 8.1.2 容易得出

$$\begin{aligned}
M_{f_{[3]}}&=\delta_2[1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 1\ 2\ 1\ 2\ 2\ 2\ 1\ 2],\\
M_{f_{[4]}}&=\delta_2[2\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 2],\\
M_{f_{[3,4]}}&=\delta_2[2\ 1\ 2\ 2\ 2\ 2\ 1\ 2].
\end{aligned}$$

直接计算可得 (8.2.7) 的代数形式为

$$Mx=1,$$

这里 $x=\ltimes_{i=1}^5 x_i$, 且

$$M=\delta_2[2\ 1\ 2\ 2\ 2\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 2\ 2\ 1\ 2\ 1\ 2].$$

因此可得, 其解为

$$\left\{x=\delta_{32}^i|i=2,6,7,23,29,31\right\},$$

或者, 其逻辑表示为

$$\begin{aligned}
\{&(1,1,1,1,0),\quad (1,1,0,0,1),\quad (1,1,0,0,0),\\
&(0,1,0,0,1),\quad (0,0,0,1,1),\quad (0,0,0,0,1)\}.
\end{aligned}$$

下面讨论布尔微分方程, 即在 (8.2.1) 中, 布尔函数 $f(x_1,\cdots,x_n)$ 未知, 但其初始值 $f(\mathbf{0})$ 以及 f 某些布尔导数的初始值 (在 $\mathbf{0}$ 点的值) 已知.

由于布尔函数由其真值向量唯一确定, 解布尔微分方程可转化为解关于未知函数真值向量的代数方程. 我们用一个例子来说明.

例 8.2.2　考虑如下布尔微分方程

$$\begin{cases}
\dfrac{\partial F}{\partial x_3}=\neg x_1\wedge\neg x_4,\\[2mm]
\dfrac{\partial^2 F}{\partial x_1\partial x_4}=\neg(x_2\vee x_3)\vee(x_2\wedge x_3),\\[2mm]
\dfrac{\partial^2 F}{\partial x_2\partial x_4}=\neg x_1,\\[2mm]
\dfrac{\partial^2 F}{\partial x_1\partial x_3}\vee\dfrac{\partial^2 F}{\partial x_1\partial x_2}=1,
\end{cases} \tag{8.2.8}$$

设初值为 $F(\mathbf{0})=0$.

在向量形式下方程变为

$$\begin{cases}\partial_{[3]}(M_F)=\delta_2[2\ 2\ 2\ 2\ 2\ 1\ 2\ 1],\\ \partial_{[1,4]}(M_F)=\delta_2[1\ 2\ 2\ 1],\\ \partial_{[2,4]}(M_F)=\delta_2[2\ 2\ 1\ 1],\\ \partial_{[1,3]}(M_F)\vee\partial_{[1,2]}(M_F)=\delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1].\end{cases}\tag{8.2.9}$$

设 M_F 的第一行为

$$m_F=[a_1\ a_2\ \cdots\ a_{16}].$$

根据 (8.2.9) 可得

$$\begin{aligned}
&a_1\langle+\rangle a_3=0, && a_2\langle+\rangle a_4=0,\\
&a_5\langle+\rangle a_7=0, && a_6\langle+\rangle a_8=0,\\
&a_9\langle+\rangle a_{11}=0, && a_{10}\langle+\rangle a_{12}=1,\\
&a_{13}\langle+\rangle a_{15}=0, && a_{14}\langle+\rangle a_{16}=1,\\
&a_1\langle+\rangle a_2\langle+\rangle a_9\langle+\rangle a_{10}=1, && a_3\langle+\rangle a_4\langle+\rangle a_{11}\langle+\rangle a_{12}=0,\\
&a_5\langle+\rangle a_6\langle+\rangle a_{13}\langle+\rangle a_{14}=0, && a_7\langle+\rangle a_8\langle+\rangle a_{15}\langle+\rangle a_{16}=1,\\
&a_1\langle+\rangle a_2\langle+\rangle a_5\langle+\rangle a_6=0, && a_3\langle+\rangle a_4\langle+\rangle a_7\langle+\rangle a_8=0,\\
&a_9\langle+\rangle a_{10}\langle+\rangle a_{13}\langle+\rangle a_{14}=1, && a_{11}\langle+\rangle a_{12}\langle+\rangle a_{15}\langle+\rangle a_{16}=1,\\
&a_3\langle+\rangle a_7\langle+\rangle a_{11}\langle+\rangle a_{15}=1, && a_4\langle+\rangle a_8\langle+\rangle a_{12}\langle+\rangle a_{16}=0.
\end{aligned}$$

因为 $F(\mathbf{0})=0$ 可知 $a_{16}=0$. 不难得到解为

$$\begin{aligned}m_F^{\mathrm{T}}&=\mathrm{Row}_1(M_F)\\ &=[\begin{matrix}a & b & a & b\\ c & a\langle+\rangle\neg b\langle+\rangle\neg c & c & a\langle+\rangle\neg b\langle+\rangle\neg c\\ \neg b\langle+\rangle\neg c & a\langle+\rangle\neg c & \neg b\langle+\rangle\neg c & a\langle+\rangle c\\ a\langle+\rangle\neg b & 1 & a\langle+\rangle\neg b & 0\end{matrix}],\end{aligned}$$

这里 a,b 和 c 可为任意布尔值. 例如,

(i) 取 $a=1$, $b=0$, 及 $c=1$, 则得

$$\begin{aligned}F(x_1,x_2,x_3,x_4)=&(x_1\wedge x_4)\vee(\neg x_1\wedge x_2\wedge x_3)\\ &\vee(\neg x_1\wedge x_2\wedge x_4)\vee(\neg x_1\wedge\neg x_2\wedge x_3\wedge\neg x_4).\end{aligned}$$

(ii) 取 $a=0$, $b=0$, 及 $c=1$, 则得

$$F(x_1,x_2,x_3,x_4)=[x_1\wedge(x_2\bar{\vee}x_4)]\vee[\neg x_1\wedge(\neg x_2\bar{\vee}x_3)\wedge\neg x_4].$$

8.3 布尔积分

至今尚无对布尔积分的统一定义, 例如, 文献 [140] 给出一种定义, 但它同时给出的微分定义又与我们前面的定义 (最常用的定义) 不相容. 本节以下给出的定义跟本章此前的微分定义相容: 即为微分的逆.

8.3.1 原函数

首先讨论一个给定布尔函数的原函数.

定义 8.3.1　给定一个布尔函数 $f(x_1,\cdots,x_n)$. 布尔函数 $F(x_1,\cdots,x_{i-1},z,x_i,\cdots,x_n)$ 称为 $f(x)$ 的第 i 个原函数, 或称为 $f(x)$ 的第 i 个偏积分, 记作

$$\int f(x_1,\cdots,x_n)\mathrm{d}[i]=F(x_1,\cdots,x_{i-1},z,x_i,\cdots,x_n), \tag{8.3.1}$$

如果

$$\frac{\partial F}{\partial z}=f(x_1,\cdots,x_n). \tag{8.3.2}$$

根据推论 8.1.1, 求原函数的问题变为解方程

$$m_F\left[\Psi^i_{n+1}\right]^{\mathrm{T}}=m_f. \tag{8.3.3}$$

为将结果表示得比较简洁, 引入以下记号: 设 $a,b\in\mathcal{B}_{2^n}$ 为两布尔向量, $1\leqslant i\leqslant n+1$. 将 a 和 b 分为 2^{i-1} 大小相等的块, 即

$$a=[a_1\ a_2\ \cdots\ a_{2^{i-1}}];\quad b=[b_1\ b_2\ \cdots\ b_{2^{i-1}}].$$

然后, 定义两个布尔向量的 i 前置和, 记作 $\dashv_{[i]}$, 为

$$a\dashv_{[i]}b=\left[b_1,a_1\langle+\rangle b_1,b_2,a_2\langle+\rangle b_2,\cdots,b_{2^{i-1}},a_{2^{i-1}}\langle+\rangle b_{2^{i-1}}\right].$$

注意, 在一般情况下 $\dashv_{[i]}$ 是不可交换的, 也就是说, 一般情况下

$$a\dashv_{[i]}b\neq b\dashv_{[i]}a.$$

利用这个记号, 可证明以下结果:

定理 8.3.1　设 $f(x_1,\cdots,x_n)$ 为一布尔函数且其真值向量为 m_f. 那么, F 是其第 i 个原函数 $(1\leqslant i\leqslant n+1)$, 当且仅当, 存在一个逻辑常数 $c\in\mathcal{B}_{2^n}$, 使得

$$m_F=m_f\dashv_{[i]}c. \tag{8.3.4}$$

证明 将 m_F 分为 2^i 个大小相同的块, 并将它们标记为

$$m_F = \left[c_{1,1}, c_{1,2}, c_{2,1}, c_{2,2}, \cdots, c_{2^{i-1},1}, c_{2^{i-1},2}\right].$$

记

$$m_f = \left[\alpha_1, \alpha_2, \cdots, \alpha_{2^{i-1}}\right].$$

则 F 为 f 的第 i 个原函数, 当且仅当

$$c_{j,1} \langle + \rangle c_{j,2} = \alpha_j, \quad j = 1, \cdots, 2^{i-1}. \tag{8.3.5}$$

对于 (8.3.5) 的每一个分量 j, 不妨设 $c_{j,1}$ 固定, 然后, 寻找 $c_{j,2}$ 满足 (8.3.5). 考虑到这一点, 我们可以任选 $c_{j,1}$, 而令 $c_{j,2} = \alpha_j \langle + \rangle c_{j,1}$, 它是 (8.3.5) 的唯一解. 结果显见. □

与普通积分类似, 多重积分可通过由里到外逐层积分得到. 我们用一个例子说明这个积分过程.

例 8.3.1 设 $f(x_1, x_2, x_3) = x_3 \wedge (x_1 \vee (x_2 \leftrightarrow x_3))$. 求

$$\int f(x_1, x_2, x_3) \mathrm{d}[2].$$

容易算出

$$m_f = [1\ 0\ 1\ 0\ 1\ 0\ 0\ 0].$$

设

$$F(x_1, z, x_2, x_3) = \int f(x_1, x_2, x_3) \mathrm{d}[2],$$

其真值向量为

$$m_F = [\alpha_1\ \alpha_2\ \alpha_3\ \alpha_4\ \alpha_5\ \alpha_6\ \alpha_7\ \alpha_8\ \alpha_9\ \alpha_{10}\ \alpha_{11}\ \alpha_{12}\ \alpha_{13}\ \alpha_{14}\ \alpha_{15}\ \alpha_{16}]^{\mathrm{T}}.$$

令

$$\alpha_5 = c_1,\ \alpha_6 = c_2,\ \alpha_7 = c_3,\ \alpha_8 = c_4,\ \alpha_{13} = c_5,\ \alpha_{14} = c_6,\ \alpha_{15} = c_7,\ \alpha_{16} = c_8.$$

可得 $\alpha_1 = c_1 \langle + \rangle (m_f)_1 = \neg c_1$. 类似计算可得

$$m_F = [\neg c_1\ c_2\ \neg c_3\ c_4\ c_1\ c_2\ c_3\ c_4\ \neg c_5\ c_6\ c_7\ c_8\ c_5\ c_6\ c_7\ c_8]^{\mathrm{T}}.$$

给定任意集合的参数, 我们可以找到相应的原函数. 因此, 我们总共有 2^8 个原系数. 为得到一个特殊的原函数, 选择一组系数, 例如, $c_1 = c_2 = c_3 = c_4 = 0$, $c_5 = c_6 = c_7 = c_8 = 1$. 那么, 有

$$m_F = [1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1]^{\mathrm{T}}.$$

于是可知

$$
\begin{aligned}
F(x_1,z,x_2,x_3) = &(x_1 \wedge z \wedge x_3) \vee ((\neg x_1) \wedge z \\
&\wedge((x_2 \wedge (\neg x_3)) \vee (\neg x_2))) \vee ((\neg x_1) \wedge (\neg z)).
\end{aligned}
$$

下面考虑多重积分的情况, 定义

$$
\begin{aligned}
&\int \cdots \int f(x_1,\cdots,x_n)\mathrm{d}[i_1]\cdots\mathrm{d}[i_k] \\
:= &\int \left(\cdots \int \left(\int f(x_1,\cdots,x_n)\mathrm{d}[i_1]\right)\mathrm{d}[i_2]\cdots\right)\mathrm{d}[i_k]. \quad (8.3.6)
\end{aligned}
$$

注意, 要让指标有意义, 规定

$$
1 \leqslant i_j \leqslant n+j, \quad j=1,\cdots,k.
$$

下面给一个例子.

例 8.3.2　设 $f(x_1,x_2,x_3)$ 如例 8.3.1 所给出. 计算

$$
\iint f(x_1,x_2,x_3)\mathrm{d}[3]\mathrm{d}[5].
$$

类似例 8.3.1, 可得出

$$
G(x_1,x_2,z_1,x_3) = \int f(x_1,x_2,x_3)\mathrm{d}[3].
$$

不难发现

$$
m_G = [c_1 \langle + \rangle 1\ c_2\ c_1\ c_2\ c_3 \langle + \rangle 1\ c_4\ c_3\ c_4\ c_5 \langle + \rangle 1\ c_6\ c_5\ c_6\ c_7\ c_8\ c_7\ c_8].
$$

接着计算

$$
F(x_1,x_2,z_1,x_3,z_2) = \int G(x_1,x_2,z_1,x_3)\mathrm{d}[5].
$$

于是可知

$$
\begin{aligned}
m_F = \ &[d_1 \langle + \rangle c_1 \langle + \rangle 1 \quad d_1 \quad d_2 \langle + \rangle c_2 \quad d_2 \quad d_3 \langle + \rangle c_1 \quad d_3 \quad d_4 \langle + \rangle c_2 \quad d_4 \\
&d_5 \langle + \rangle c_3 \langle + \rangle 1 \quad d_5 \quad d_6 \langle + \rangle c_4 \quad d_6 \quad d_7 \langle + \rangle c_3 \quad d_7 \quad d_8 \langle + \rangle c_4 \quad d_8 \\
&d_9 \langle + \rangle c_5 \langle + \rangle 1 \quad d_9 \quad d_{10} \langle + \rangle c_6 \quad d_{10} \quad d_{11} \langle + \rangle c_5 \quad d_{11} \quad d_{12} \langle + \rangle c_6 \quad d_{12} \\
&d_{13} \langle + \rangle c_7 \quad d_{13} \quad d_{14} \langle + \rangle c_8 \quad d_{14} \quad d_{15} \langle + \rangle c_7 \quad d_{15} \quad d_{16} \langle + \rangle c_8 \quad d_{16}],
\end{aligned}
$$

这里 c_i, $i=1,\cdots,8$, d_j, $j=1,\cdots,16$ 为任意布尔数.

例如, 我们可以有一些特殊的 F 函数如下.

(i) 设 $c_i=0$, $\forall i$, 以及 $d_j=0$, $\forall j$. 那么

$$
F(x_1,\cdots,x_5) = (x_1 \vee x_2) \wedge x_3 \wedge x_4 \wedge x_5.
$$

(ii) 设 $c_i=0$, $\forall i$, $d_1=d_5=d_9=d_{16}=1$, 且 $d_j=0$, $j \neq 1,5,9,16$. 那么

$$
F(x_1,\cdots,x_5) = [(x_1 \vee x_2) \wedge x_3 \wedge x_4 \wedge \neg x_5] \vee [\neg(x_1 \vee x_2 \vee x_3 \vee x_4)].
$$

8.3.2 不定积分

定义 8.3.2 给定一个逻辑函数 $F(x_1,\cdots,x_n)$. 其微分形式, 记为 $\mathrm{d}F$, 定义为

$$\mathrm{d}F := \frac{\partial F}{\partial x_1}\mathrm{d}x_1 + \cdots + \frac{\partial F}{\partial x_n}\mathrm{d}x_n. \tag{8.3.7}$$

注意, 在 (8.3.7) 式中符号 "+" 不是一个算子, 它只表示一种邻接关系.

定义 8.3.3 给定一组函数

$$f_i(x_1,\cdots,x_{i-1},\hat{x}_i,x_{i+1}\cdots,x_n), \quad i=1,\cdots,n.$$

一个函数 $F(x_1,\cdots,x_n)$ 称为微分形式

$$\mathrm{d}h = f_1\mathrm{d}x_1 + f_2\mathrm{d}x_2 + \cdots + f_n\mathrm{d}x_n$$

的不定积分 (或简称为 $\{f_1,\cdots,f_n\}$ 的不定积分), 如果

$$\frac{\partial F}{\partial x_i} = f_i, \quad i=1,\cdots,n. \tag{8.3.8}$$

由等式 (8.1.10) 可知, 如果 F 是 $\mathrm{d}h$ 的不定积分, 那么 $\bar{F}$ 也是.

下面考虑, 什么时候不定积分存在.

定理 8.3.2 考察一个微分形式

$$\begin{aligned}\mathrm{d}h = {} & h_1(\hat{x}_1,x_2,\cdots,x_n)\mathrm{d}x_1 + h_2(x_1,\hat{x}_2,\cdots,x_n)\mathrm{d}x_2 \\ & + \cdots + h_n(x_1,x_2\cdots\hat{x}_n)\mathrm{d}x_n.\end{aligned}$$

那么, 至少存在一对互余的不定积分, 当且仅当

$$\frac{\partial h_i}{\partial x_j} = \frac{\partial h_j}{\partial x_i}, \quad 1 \leqslant i < j \leqslant n. \tag{8.3.9}$$

证明 必要性是显然的, 证明充分性. 利用 $\{h_i|i=1,\cdots,n\}$, 可计算

$$\frac{\partial h_i}{\partial x_j} = \frac{\partial h_j}{\partial x_i}, \quad 1 \leqslant i < j \leqslant n.$$

类似地, 可以计算二阶交叉导数

$$\frac{\partial^2 h_i}{\partial x_j \partial x_k} = \frac{\partial^2 h_j}{\partial x_i \partial x_k} = \frac{\partial^2 h_k}{\partial x_i \partial x_j},$$

以及更高阶的交叉导数. 利用得到的交叉导数以及 MacLaurin 展式, 可以构造

$$\begin{aligned}F(x_1,\cdots,x_n) = {} & c\,\langle+\rangle \bigoplus_{i=1}^{n} h_i|_{\mathbf{0}} \wedge x_i \,\langle+\rangle \bigoplus_{1\leqslant i_1<i_2\leqslant n} \left.\frac{\partial h_{i_1}}{\partial x_{i_2}}\right|_{\mathbf{0}} \wedge x_{i_1} \wedge x_{i_2} \\ & \langle+\rangle \bigoplus_{1\leqslant i_1<i_2<i_3\leqslant n} \left.\frac{\partial^2 h_{i_1}}{\partial x_{i_2}\partial x_{i_3}}\right|_{\mathbf{0}} \wedge x_{i_1} \wedge x_{i_2} \wedge x_{i_3} \\ & \langle+\rangle \cdots \langle+\rangle \left.\frac{\partial^{n-1} h_1}{\partial x_2 \cdots \partial x_n}\right|_{\mathbf{0}} \wedge x_1 \wedge x_2 \wedge \cdots \wedge x_n.\end{aligned} \tag{8.3.10}$$

现在, 不难证明

$$F(x) = \int \mathrm{d}h. \qquad \square$$

下面的结果来自定理 8.3.2 的构造性证明.

推论 8.3.1 如果 $\int \mathrm{d}h$ 存在, 那么, 它是在互余意义下唯一 (即, 只容许它的余函数同为 h 的不定积分.)

在下面讨论微分形式的不定积分时, 总假定:

A1 可积性条件 (8.3.9) 成立.

这样, 当 $\mathrm{d}h$ 可积时, 我们将不定积分写成

$$\int \mathrm{d}h = F(x) \langle + \rangle c,$$

这里, $c \in \mathcal{D}$.

在以后的应用中, 我们希望确指 F. 所以规定以下记号:

$$\int \mathrm{d}h = F(x), \quad F(0) = 0;$$

$$\int \bar{\mathrm{d}}h = \bar{F}(x), \quad \bar{F}(0) = 1.$$

下面讨论如何计算不定积分. 实际上, 定理 8.3.2 的构造性证明已经给了一个算法. 我们讨论另一个简单算法.

下面的结果是推论 8.1.1 的一个直接应用.

定理 8.3.3 微分形式 $\mathrm{d}h = f_1 \mathrm{d}x_1 + \cdots + f_n \mathrm{d}x_n$ 的每一个不定积分具有一个如下的线性 Galois 代数方程的解 z 作为它的真值向量.

$$\Psi_n \langle \ltimes \rangle z = b, \tag{8.3.11}$$

这里,

$$\Psi_n = \begin{bmatrix} \Psi_n^1 \\ \Psi_n^2 \\ \vdots \\ \Psi_n^n \end{bmatrix} \in \mathcal{B}_{n2^{n-1} \times 2^n}; \quad 以及 \quad b = \begin{bmatrix} m_{f_1}^{\mathrm{T}} \\ m_{f_2}^{\mathrm{T}} \\ \vdots \\ m_{f_n}^{\mathrm{T}} \end{bmatrix} \in \mathcal{B}_{n2^{n-1}}.$$

值得一提的是, 要得到约定解, 应令 z 的最后一个分量为零, 即, $z_{2^n} = 0$, 这对应 $F(0) = 0$.

例 8.3.3 1) 设 $n = 2$. 那么有

$$\Psi_2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

情况 1: 设 $f_1 = x_2$, $f_2 = \neg x_1$. 那么有 $m_{f_1} = [1\ 0]$, $m_{f_2} = [0\ 1]$. 等式 (8.3.11) 变为

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \langle \ltimes \rangle \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

等价地, 有

$$\begin{cases} z_1 \langle + \rangle z_3 = 1, \\ z_2 \langle + \rangle z_4 = 0, \\ z_1 \langle + \rangle z_2 = 0, \\ z_3 \langle + \rangle z_4 = 1. \end{cases}$$

令 $z_4 = 0$, 也就是令 $F(\mathbf{0}) = 0$, 有

$$\begin{cases} z_1 = 0, \\ z_2 = 0, \\ z_3 = 1, \\ z_4 = 0. \end{cases}$$

即, $m_F = [0\ 0\ 1\ 0]$. 因此, $F = (\neg x_1) \wedge x_2$. 写成积分形式, 可得

$$\int x_2 \mathrm{d}x_1 + \neg x_1 \mathrm{d}x_2 = (\neg x_1) \wedge x_2. \tag{8.3.12}$$

我们还有互余意义下的解

$$\int x_2 \bar{\mathrm{d}} x_1 + \neg x_1 \bar{\mathrm{d}} x_2 = (\neg x_1) \wedge x_2 \langle + \rangle 1. \tag{8.3.13}$$

情况 2: 设 $f_1 = x_2$, $f_2 = 1$. 那么, 有 $m_{f_1} = [1\ 0]$ 及 $m_{f_2} = [1\ 1]$. 容易检验, 此时无解, 即不定积分不存在.

2) 假定 $n = 3$. 那么, 不难算得

$$\Psi_3 = \begin{bmatrix} 1&0&0&0&1&0&0&0\\ 0&1&0&0&0&1&0&0\\ 0&0&1&0&0&0&1&0\\ 0&0&0&1&0&0&0&1\\ 1&0&1&0&0&0&0&0\\ 0&1&0&1&0&0&0&0\\ 0&0&0&0&1&0&1&0\\ 0&0&0&0&0&1&0&1\\ 1&1&0&0&0&0&0&0\\ 0&0&1&1&0&0&0&0\\ 0&0&0&0&1&1&0&0\\ 0&0&0&0&0&0&1&1 \end{bmatrix}.$$

考察

$$\mathrm{d}h = (x_3 \to x_2)\mathrm{d}x_1 + \neg(x_1 \wedge x_3)\mathrm{d}x_2 + (x_1 \wedge (\neg x_2))\mathrm{d}x_3$$

的不定积分. 容易算得

$$m_{f_1} = [1\ 1\ 0\ 1]; \quad m_{f_2} = [0\ 1\ 1\ 1]; \quad m_{f_3} = [0\ 1\ 0\ 0].$$

考虑

$$\Psi_3 z = b,$$

这里,

$$b = [1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 0]^{\mathrm{T}}.$$

等价地, 有

$$\begin{cases} z_1 \langle + \rangle z_5 = z_2 \langle + \rangle z_6 = z_4 \langle + \rangle z_8 = z_2 \langle + \rangle z_4 \\ \qquad = z_5 \langle + \rangle z_7 = z_6 \langle + \rangle z_8 = z_3 \langle + \rangle z_4 = 1, \\ z_3 \langle + \rangle z_7 = z_1 \langle + \rangle z_3 = z_1 \langle + \rangle z_2 = z_5 \langle + \rangle z_6 = z_7 \langle + \rangle z_8 = 0. \end{cases}$$

令 $z_8 = 0$, 有如下解:

$$m_F = [0\ 0\ 0\ 1\ 1\ 1\ 0\ 0].$$

容易算出其相应函数, 于是有

$$\begin{aligned} &\int (x_3 \to x_2)\mathrm{d}x_1 + \neg(x_1 \wedge x_3)\mathrm{d}x_2 + (x_1 \wedge (\neg x_2))\mathrm{d}x_3 \\ &= \neg(x_1 \leftrightarrow x_2) \wedge (\neg x_1 \vee \neg x_3), \end{aligned} \tag{8.3.14}$$

以及互余解

$$\begin{aligned}&\int (x_3 \to x_2)\bar{\mathrm{d}}x_1 + \neg(x_1 \wedge x_3)\bar{\mathrm{d}}x_2 + (x_1 \wedge (\neg x_2))\bar{\mathrm{d}}x_3 \\ =& (x_1 \leftrightarrow x_2) \vee (x_1 \wedge x_3).\end{aligned} \tag{8.3.15}$$

8.3.3 定积分

设 $\int \mathrm{d}h$ 的不定积分存在, 记其为 $(F, \bar{F})$. 那么, 其定积分定义如下.

定义 8.3.4 设一个微分形式 $\mathrm{d}h$ 定义如下

$$\begin{aligned}\mathrm{d}h = & f_1(\hat{x}_1, \cdots, x_n)\mathrm{d}x_1 + f_2(x_1, \hat{x}_2, \cdots, x_n)\mathrm{d}x_2 \\ &+ \cdots + f_n(x_1, \cdots, x_{n-1}, \hat{x}_n)\mathrm{d}x_n,\end{aligned}$$

同时, 一个子集 $S \subset \mathcal{D}^n$, 以及一个逻辑函数 $g(x)$ 给定.

设 $\int \mathrm{d}h = F(x)$(且 $F(0) = 0$). 那么, 定义定积分如下

$$\int_S g(x)\mathrm{d}h = \sum_{x \in S} g(x) \wedge F(x). \tag{8.3.16}$$

(注意, 这里 $\sum$ 表示 $\mathbb{R}$ 上普通意义的求和.)

同时, 对 $\bar{F}$ 我们亦可定义定积分为

$$\int_S g(x)\bar{\mathrm{d}}h = \sum_{x \in S} g(x) \wedge \bar{F}(x). \tag{8.3.17}$$

这里, S 称为积分域, $g(x)$ 称为被积函数.

容易检验, 定义 8.3.4 给出的定积分, 满足 "定积分" 的一些基本性质. 例如

1) 如果 $f(x) \leqslant g(x)$, 那么

$$\int_S f(x)\mathrm{d}h \leqslant \int_S g(x)\mathrm{d}h. \tag{8.3.18}$$

2) 如果 $S_1 \subseteq S_2$, 那么

$$\int_{S_1} g(x)\mathrm{d}h \leqslant \int_{S_2} g(x)\mathrm{d}h. \tag{8.3.19}$$

3)

$$\int_{S_1 \cup S_2} g(x)\mathrm{d}h = \int_{S_1} g(x)\mathrm{d}h + \int_{S_2} g(x)\mathrm{d}h - \int_{S_1 \cap S_2} g(x)\mathrm{d}h. \tag{8.3.20}$$

定义

$$\operatorname{supp}(f)=\{x|f(x)\neq 0\}.$$

设 $F=\int \mathrm{d}h$. 则

$$\int_S g(x)\mathrm{d}h=|\operatorname{supp}(F)\cap\operatorname{supp}(g)\cap S|.$$

例 8.3.4　回忆例 8.3.3. 我们利用相应的不定积分计算定积分.

1) $n=2$.

情况 1: 设 $f_1=x_2$, $f_2=\neg x_1$. 利用 (8.3.10), 那么, 考虑约定的定积分. 有 $F=x_2\wedge(\neg x_1)$. 设 $S=\{(x_1,x_2)|x_1\to x_2=1\}$. 则

$$S=\{(1,1),(0,1),(0,0)\}.$$

设被积函数为 $g(x_1,x_2)=x_1\leftrightarrow x_2$. 容易算得

$$\int_S (x_1\leftrightarrow x_2)x_2\mathrm{d}x_1+\neg x_1\mathrm{d}x_2=0;$$

且

$$\int_S (x_1\leftrightarrow x_2)x_2\bar{\mathrm{d}}x_1+\neg x_1\bar{\mathrm{d}}x_2=2.$$

情况 2: 设 $f_1=x_2$, $f_2=1$. 那么

$$\int_S x_2\mathrm{d}x_1+\mathrm{d}x_2$$

不存在.

2) 考虑 $n=3$ 并且

$$\mathrm{d}h=(x_3\to x_2)\mathrm{d}x_1+\neg(x_1\wedge x_3)\mathrm{d}x_2+(x_1\wedge\neg x_2)\mathrm{d}x_3.$$

设

$$S=\left\{x\in\mathcal{D}^3\middle|(x_1\vee x_2)\wedge x_3=0\right\}.$$

那么

$$S=\{(1,1,0),\ (1,0,0),\ (0,1,0),\ (0,0,0),\ (0,0,1)\}.$$

令 $g(x)=x_1\vee(x_3\to x_2)$. 利用 (8.3.14), 直接计算可得

$$\int[x_1\vee(x_3\to x_2)](x_3\to x_2)\mathrm{d}x_1+\neg(x_1\wedge x_3)\mathrm{d}x_2+(x_1\wedge(\neg x_2))\,\mathrm{d}x_3=2.$$

8.4 注释与参考

在 G. Boole 于 1847 年提出后来被称为布尔代数的理论之后不久, 对布尔微积分的研究就开始了[69]. 但是直到四十年后当 Shannon 提出切换代数在切换线路设计中的应用后, 人们才发现布尔导数在切换理论中的重要性[17, 124]. 此后有了大量对布尔微积分及其应用的深入探索, 如文献 [33, 71, 120, 131, 136, 147] 等.

本章讨论布尔微积分. 在介绍了布尔微分和积分的基本计算公式后, 讨论了解布尔代数方程及布尔微分方程的问题. 此外, 还介绍了它在组合线路的故障检测上的应用. 本章的内容主要基于文献 [55].

第 9 章　逻辑动态系统

9.1　布尔网络

一个离散时间的动态系统，设其状态变量为 $x=(x_1,\cdots,x_n)$. 如果它的状态空间是一个有限集，则称其为一个逻辑动态系统. 设 $x_i\in\mathcal{D}_{k_i}$, $i=1,\cdots,n$, 那么，这个逻辑动态系统就可以表示为

$$\begin{cases}x_1(t+1)=f_1(x_1(t),\cdots,x_n(t)),\\x_2(t+1)=f_2(x_1(t),\cdots,x_n(t)),\\\quad\vdots\\x_n(t+1)=f_n(x_1(t),\cdots,x_n(t)),\end{cases}\tag{9.1.1}$$

这里，$f_i:\ \prod_{j=1}^n\mathcal{D}_{k_j}\to\mathcal{D}_{k_i}$, $i=1,\cdots,n$ 为多值逻辑函数. (9.1.1) 称为一个多值逻辑动态系统. 如果 $k_i=k$, $i=1,\cdots,n$, 则称其为一个 k 值逻辑动态系统. 当 $k=2$ 时，一个 2 值逻辑动态系统也称为一个布尔网络. 一个多值 (k 值) 逻辑动态系统也称为多值 (k 值) 逻辑网络. 布尔网络是最简单的逻辑动态系统，或许，也可以说它是最重要的逻辑动态系统.

布尔网络真正的模型是一个物理网络，它可以用一个网络图来描述. 例如，图 9.1.1 是三个结点的一个布尔网络. 结点 x_1,x_2,x_3 在每个时刻 t 可取不同的逻辑值. 每个结点在 $t+1$ 时刻的值是它的邻域结点在 t 时刻值的一个逻辑函数. 网络图是一个有向图，其每个结点的入度是指指向它的边的个数，其出度是指从它出发的边的个数. 网络图通常只画出邻域关系，这时，我们将决定某一结点“入度”的那些结点称为该点的邻域. 例如: 图 9.1.1 中，x_1 的邻域为 $\{x_2,x_3\}$, 记作 $U(x_1)=\{x_2,x_3\}$; x_2 的邻域为 $U(x_2)=\{x_1\}$; x_3 的邻域为 $U(x_3)=\{x_2,x_3\}$. 点与其邻域的逻辑值之间的关系，有时也会标在网络图上. 但当网络较复杂时，这种方法是很不方便的. 于是，人们常用单独的逻辑函数来表示网络结点的逻辑动态演化过程，这就是逻辑动态方程. 例如，图 9.1.1 中的布尔网络的一个可能的动态方程为 (9.1.2).

$$\begin{cases}x_1(t+1)=x_2(t)\bar{\vee}x_3(t),\\x_2(t+1)=\neg x_1(t),\\x_3(t+1)=x_2(t)\wedge(x_3(t)\to x_2(t)).\end{cases}\tag{9.1.2}$$

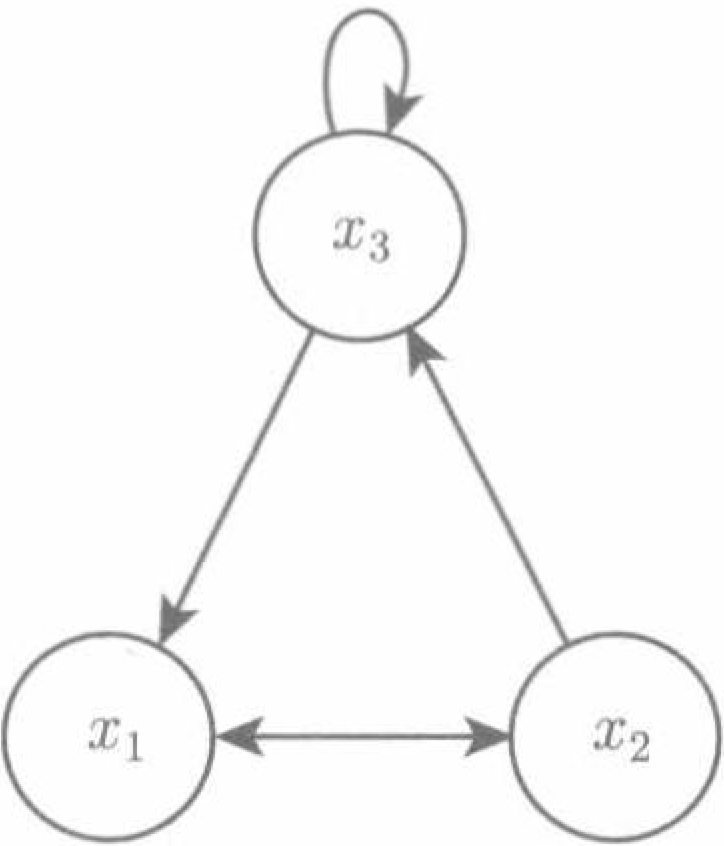


图 9.1.1 布尔网络

直接讨论一个逻辑形式的演化方程是很困难的. 我们的目的是利用逻辑变量的矩阵表示将其转化为代数状态空间方程. 在代数状态空间方程中只有普通的矩阵乘法, 因此, 对逻辑动态系统的分析与综合都很方便. 下面详述这个转化的四个步骤:

第一步, 逻辑变量与逻辑算子的矩阵表示.

利用 $0 \sim \delta_2^2$ 及 $1 \sim \delta_2^1$ 将逻辑变量用向量表示, 再利用第 2 章和第 3 章中标准逻辑算子的结构矩阵 (表 9.1.1) 则可将逻辑表达式变为代数表达式.

表 9.1.1 逻辑算子的结构矩阵

算子	结构矩阵	算子	结构矩阵
$\neg$	$M_n = \delta_2[2\ 1]$	$\vee$	$M_d = \delta_2[1\ 1\ 1\ 2]$
$\wedge$	$M_c = \delta_2[1\ 2\ 2\ 2]$	$\rightarrow$	$M_i = \delta_2[1\ 2\ 1\ 1]$
$\leftrightarrow$	$M_e = \delta_2[1\ 2\ 2\ 1]$	$\bar{\vee}$	$M_{\bar{\vee}} = \delta_2[2\ 1\ 1\ 2]$
$\uparrow$	$M_{\uparrow} = \delta_2[2\ 1\ 1\ 1]$	$\downarrow$	$M_{\downarrow} = \delta_2[2\ 2\ 2\ 1]$

以 (9.1.2) 为例, 它可变为

$$\begin{cases} x_1(t+1) = M_{\bar{\vee}} x_2(t) x_3(t), \\ x_2(t+1) = M_n x_1(t), \\ x_3(t+1) = M_c x_2(t) \left(M_i x_3(t) x_2(t)\right). \end{cases} \tag{9.1.3}$$

第二步, 换序与降阶.

这时, 可能要反复使用以下三个工具:

• 用换位矩阵

$$W_{[m,n]} XY = YX, \quad X \in \mathbb{R}^m,\ Y \in \mathbb{R}^n$$

将因子按增序排好;

• 用公式 (1.3.8)

$$XM = (I_t \otimes M) X, \quad X \in \mathbb{R}^t$$

将常数矩阵 M 与变量 X 换位.

- 用降阶矩阵

$$M_r := \begin{bmatrix} \delta_2^1 & 0 \\ 0 & \delta_2^2 \end{bmatrix}$$

将 x^s 的幂次降为 1(见命题 3.1.1).

考察 (9.1.3) 的第 3 个方程, 有

$$\begin{aligned} x_3(t+1) &= M_c x_2(t) M_i W_{[2]} x_2(t) x_3(t) \\ &= M_c \left(I_2 \otimes M_i W_{[2]}\right) x_2(t)^2 x_3(t) \\ &= M_c \left(I_2 \otimes M_i W_{[2]}\right) M_r x_2(t) x_3(t) \\ &= \tilde{M}_3 x_2(t) x_3(t), \end{aligned}$$

这里

$$\begin{aligned} \tilde{M}_3 &= M_c \left(I_2 \otimes M_i W_{[2]}\right) M_r \\ &= \delta_2[1\ 1\ 2\ 2]. \end{aligned}$$

(9.1.3) 变为

$$\begin{cases} x_1(t+1) = M_{\triangledown} x_2(t) x_3(t), \\ x_2(t+1) = M_n x_1(t), \\ x_3(t+1) = \tilde{M}_3 x_2(t) x_3(t). \end{cases} \tag{9.1.4}$$

第三步, 补齐哑变量.

为了得到整个系统的代数状态空间方程, 我们需要每个方程都含全部变量, 因此, 需要补充一些哑变量, 如 (9.1.4) 的第 1 和第 3 个方程需补上 $x_1(t)$, 第 2 个方程需补上 $x_2(t)$ 和 $x_3(t)$. 为此, 需要一些准备.

命题 9.1.1　设 $x \in \Delta_m$, $y \in \Delta_n$, $z \in \Delta_r$. 定义

$$\begin{aligned} F_{[m,n,r]} &:= I_m \otimes \mathbf{1}_{nr}^{\mathrm{T}}; \\ M_{[m,n,r]} &:= \mathbf{1}_m^{\mathrm{T}} \otimes I_n \otimes \mathbf{1}_r^{\mathrm{T}}; \\ R_{[m,n,r]} &:= \mathbf{1}_{mn}^{\mathrm{T}} \otimes I_r. \end{aligned}$$

那么

$$\begin{aligned} F_{[m,n,r]} xyz &= x; \\ M_{[m,n,r]} xyz &= y; \\ R_{[m,n,r]} xyz &= z. \end{aligned} \tag{9.1.5}$$

如果只有两个因子, 则有

推论 9.1.1 设 $x \in \Delta_m$, $y \in \Delta_n$. 定义

$$
\begin{aligned}
F_{[m,n]} &:= I_m \otimes \mathbf{1}_n^{\mathrm{T}};\\
R_{[m,n]} &:= \mathbf{1}_m^{\mathrm{T}} \otimes I_n.
\end{aligned}
$$

那么

$$
\begin{aligned}
F_{[m,n]}xy &= x;\\
R_{[m,n]}xy &= y.
\end{aligned} \tag{9.1.6}
$$

这些等式很容易证明, 这里留作练习. 利用它们, 我们就可以自由地添加哑变量了.

考察 (9.1.4), 利用 (9.1.6) 可得

$$
\begin{cases}
x_1(t+1) = M_{\triangledown} R_{[2,4]} x_1(t) x_2(t) x_3(t) := M_1 x(t),\\
x_2(t+1) = M_n F_{[2,4]} x_1(t) x_2(t) x_3(t) := M_2 x(t),\\
x_3(t+1) = \tilde{M}_3 R_{[2,4]} x_1(t) x_2(t) x_3(t) := M_3 x(t),
\end{cases} \tag{9.1.7}
$$

这里 $x(t) = \ltimes_{i=1}^{3} x_i(t)$,

$$
\begin{aligned}
M_1 &= M_{\triangledown} R_{[2,4]} = \delta_2[2\ 1\ 1\ 2\ 2\ 1\ 1\ 2],\\
M_2 &= M_n F_{[2,4]} = \delta_2[2\ 2\ 2\ 2\ 1\ 1\ 1\ 1],\\
M_3 &= \tilde{M}_3 R_{[2,4]} = \delta_2[1\ 1\ 2\ 2\ 1\ 1\ 2\ 2].
\end{aligned}
$$

第四步, 生成代数状态空间方程.

回忆第 3 章, 我们曾经证明: 如果

$$
\begin{cases}
x_1(t+1) = M_1 x(t),\\
x_2(t+1) = M_2 x(t),\\
\quad\vdots\\
x_n(t+1) = M_n x(t),
\end{cases}
$$

这里, $x(t) = \ltimes_{i=1}^{n} x_i(t)$, 那么, 系统的代数状态空间方程为

$$
x(t+1) = Mx(t), \tag{9.1.8}
$$

这里

$$
M = M_1 * M_2 * \cdots * M_n.
$$

利用 (9.1.7), 则布尔网络 (9.1.2) 有形如 (9.1.8) 的代数状态空间方程为其状态演化方程, 其中

$$\begin{aligned} M &= M_1 * M_2 * M_3 \\ &= \delta_8[7\ 3\ 4\ 8\ 5\ 1\ 2\ 6]. \end{aligned} \tag{9.1.9}$$

9.2 控制布尔网络

一个布尔网络, 如果一些结点, 其值可以依设计要求任意选择, 则将这样的结点称为输入或控制, 如果还有一些结点, 它们的值不影响网络演化, 则这些结点称为输出. 带有输入–输出的布尔网络称为控制布尔网络.

一个 n 个结点的控制布尔网络的动态方程可表示成

$$\begin{cases} x_1(t+1) = f_1(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \\ x_2(t+1) = f_2(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \end{cases} \tag{9.2.1}$$
$$y_j(t) = g_j(x_1(t),\cdots,x_n(t)), \quad j = 1,\cdots,p.$$

这里 $u_i(t),\ i=1,\cdots,m$ 是控制, $y_j(t),\ j=1,\cdots,p$ 是输出.

利用逻辑变量的向量表示, 类似布尔网络, 也可以得到控制布尔网络 (9.2.1) 的代数状态空间表示:

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{9.2.2}$$

这里 $x(t) = \ltimes_{i=1}^n x_i(t)$, $u(t) = \ltimes_{i=1}^m u_i(t)$, $y(t) = \ltimes_{i=1}^p y_i(t)$, $L \in \mathcal{L}_{2^n\times 2^{n+m}}$, $H \in \mathcal{L}_{2^p\times 2^n}$.

下面给一个例子.

例 9.2.1 设一控制布尔网络的网络图如图 9.2.1. 其动态方程为

$$\begin{cases} A(t+1) = B(t) \vee u_1(t) \\ B(t+1) = \neg C(t) \leftrightarrow u_2(t) \\ C(t+1) = \neg A(t), \end{cases} \tag{9.2.3}$$
$$y(t) = B(t) \wedge C(t).$$

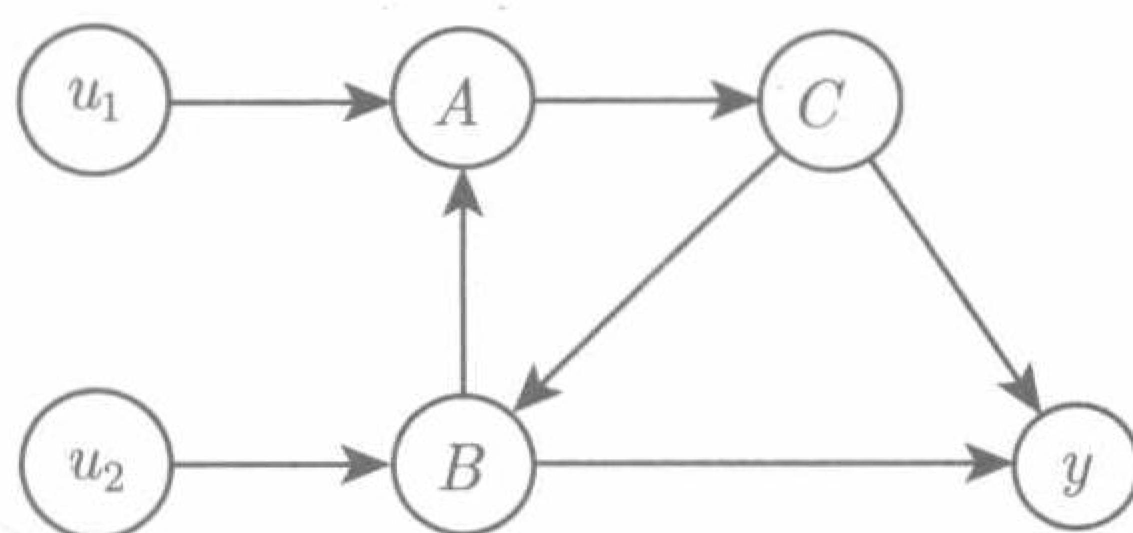


图 9.2.1 布尔控制网络

那么, 有

$$\begin{aligned}
A(t+1) &= M_d u_1(t) B(t) \\
&= M_d D_f[2,2](t) M_{[2,2,2]} x(t) \\
&= M_d D_f[2,2]\left(I_4 \otimes M_{[2,2,2]}\right) u(t) x(t) \\
&:= L_1 u(t) x(t). \\
B(t+1) &= M_e M_n u_2(t) C(t) \\
&= M_e M_n D_r[2,2] u(t) D_r[4,2] x(t) \\
&= M_e M_n D_r[2,2]\left(I_4 \otimes D_r[4,2]\right) u(t) x(t) \\
&:= L_2 u(t) x(t). \\
C(t+1) &= M_n A(t) \\
&= M_n D_f[2,4] x(t) \\
&= M_n D_f[2,4] D_r[4,2] u(t) x(t) \\
&:= L_3 u(t) x(t). \\
y(t) &= M_c B(t) C(t) \\
&= M_c D_r[2,2] x(t) \\
&:= H x(t).
\end{aligned}$$

于是可得

$$\begin{aligned}
L_1 =& M_d D_f[2,2]\left(I_4 \otimes D_m[2,2,2]\right) \\
=& \delta_2[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1, \\
& 1,1,2,2,1,1,2,2,1,1,2,2,1,1,2,2]. \\
L_2 =& M_e M_n D_r[2,2]\left(I_4 \otimes D_r[4,2]\right) \\
=& \delta_2[2,1,2,1,2,1,2,1,1,2,1,2,1,2,1,2, \\
& 2,1,2,1,2,1,2,1,1,2,1,2,1,2,1,2]. \\
L_3 =& M_n D_f[2,4] D_r[4,2] \\
=& \delta_2[2,2,2,2,1,1,1,1,2,2,2,2,1,1,1,1, \\
& 2,2,2,2,1,1,1,1,2,2,2,2,1,1,1,1].
\end{aligned}$$

最后, 得到系统 (9.2.3) 的代数状态空间表达式 (9.2.2), 其中 $x(t)=A(t)B(t)C(t)$,

$u(t) = u_1(t)u_2(t)$, 以及

$$
\begin{aligned}
L &= L_1 * L_2 * L_3 \\
&= \delta_8[4,2,4,2,3,1,3,1,2,4,2,4,1,3,1,3, \\
&\quad 4,2,8,6,3,1,7,5,2,4,6,8,1,3,5,7], \\
H &= M_c D_r[2,2] = \delta_2[1,2,2,2,1,2,2,2].
\end{aligned}
$$

9.3　布尔网络的拓扑结构

定义 9.3.1　考虑布尔网络 (9.1.1). x_e 称为一个不动点 (fixed point), 如果 $x(t) = x_e$ 则得 $x(s) = x_e, \forall s > t$. $(x_0, x_1, \cdots, x_\ell = x_0)$ 称为一个极限环 (cycle) 如果 $x(t) = x_0$ 则有 $x(t+j) = x_j, j = 1, \cdots, \ell$. 设 $x_i \neq x_j, 0 \leqslant i < j \leqslant \ell - 1$. 则称 ℓ 为极限环的长度. 不动点与极限环统称为吸引子 (attractor).

因为布尔网络只有有限个点 (设为 n), 而每个点只有两个状态, 所以总共只有 2^n 个状态. 在向量形式下, $x(t) \in \Delta_{2^n}$. 因此, 至多经过 2^n 个时刻, 必出现重复状态. 换言之, 布尔网络的轨线一定要收敛到一个吸引子. 因此, 不动点和极限环是布尔网络最重要的拓扑结构.

定理 9.3.1　考虑布尔网络 (9.1.1), 其代数表达式为 (9.1.8).

(i) $\delta_{2^n}^i$ 是其不动点, 当且仅当, (9.1.8) 中 L 的对角线元素

$$
\ell_{ii} = 1.
$$

(ii) 布尔网络 (9.1.1) 的不动点数, 记为 N_e, 为

$$
N_e = \mathrm{Trace(L)}. \tag{9.3.1}
$$

下面考虑极限环. 首先引入一个记号: 设 $s, k \in \mathbb{Z}_+$ 为两个正整数. 如果 $k < s$ 且 $\frac{s}{k} \in \mathbb{Z}_+$. 则 k 称为 s 的真因子. s 的真因子集合记为 $\mathcal{P}(s)$. 例如, $\mathcal{P}(8) = \{1,2,4\}$, $\mathcal{P}(12) = \{1,2,3,4,6\}$, 等.

注意到, 如果 x 在一个长度为 ℓ 的极限环上, 则 x 为 L^ℓ 的不动点. 利用定理 9.3.1, 再考虑到可能的重复, 则不难得到如下的结论.

定理 9.3.2　布尔网络 (9.1.1) 的长度为 s 的极限环数, 记作 N_s, 可由下式递推得到

$$
\begin{cases}
N_1 = N_e, \\
N_s = \dfrac{\mathrm{Trace}(L^s) - \sum\limits_{k \in \mathcal{P}(s)} k N_k}{s}, \quad 2 \leqslant s \leqslant 2^n.
\end{cases} \tag{9.3.2}
$$

下面讨论如何找到极限环. 如果

$$\mathrm{Trace}(L^s) - \sum_{k\in\mathcal{P}(s)} kN_k > 0, \tag{9.3.3}$$

则称 “s” 为非平凡指数.

设 s 为非平凡指数. 记 ℓ_{ii}^s 为 L^s 的第 i 个对角元素. 定义

$$C_s = \{i \mid \ell_{ii}^s = 1\}, \quad s = 1, 2, \cdots, 2^n,$$

并且

$$D_s = C_s \bigcap_{i\in\mathcal{P}(s)} C_i^c,$$

这里 C_i^c 是 C_i 的余集.

利用这些记号, 不难得到

命题 9.3.1 设 $x_0 = \delta_{2^n}^i$. 那么, $\{x_0, Lx_0, \cdots, L^s x_0\}$ 为长度为 s 的极限环, 当且仅当 $i \in D_s$.

利用定理 9.3.1 就可找到一个布尔网络的所有不动点. 利用定理 9.3.2 及命题 9.3.1 就可以找到所有的极限环. 具体步骤可见如下例子.

例 9.3.1 考察系统

$$\begin{cases} x_1(t+1) = x_2(t) \wedge x_3(t), \\ x_2(t+1) = \neg x_1(t), \\ x_3(t+1) = x_2(t) \vee x_3(t). \end{cases} \tag{9.3.4}$$

不难算出, 其向量形式为

$$\begin{cases} x_1(t+1) = \delta_2[1,2,2,2,1,2,2,2]x(t) \\ x_2(t+1) = \delta_2[2,2,2,2,1,1,1,1]x(t) \\ x_3(t+1) = \delta_2[1,1,1,2,1,1,1,2]x(t). \end{cases} \tag{9.3.5}$$

于是有代数状态空间方程

$$x(t+1) = Lx(t), \tag{9.3.6}$$

这里

$$L = M_1 * M_2 * M_3 = \delta_8[3,7,7,8,1,5,5,6].$$

容易检验

$$\mathrm{Trace}(L^s) = \begin{cases} 0, & s < 4, \\ 4, & s \geqslant 4. \end{cases}$$

由定理 9.3.1 可知, 系统没有不动点. 由定理 9.3.2 可知, 系统只有一个长度为 4 的极限环. 计算得

$$L^4 = \delta_8[1, 3, 3, 1, 5, 7, 7, 3].$$

根据命题 9.3.1, 任选一个对角元为 1 的列 (显然可选 $\mathrm{Col}_i(L^4)$, $i = 1, 3, 5, 7$), 设选 $Z = \mathrm{Col}_1(L^4) = \delta_8^1$, 于是有

$$LZ = \delta_8^3, \quad L^2 Z = \delta_8^7, \quad L^3 Z = \delta_8^5, \quad L^4 Z = Z.$$

即

$$\delta_8^1 \to \delta_8^3 \to \delta_8^7 \to \delta_8^5 \to \delta_8^1$$

为一极限环. 返回到 (x_1, x_2, x_3) 的布尔值, 则极限环可表示成

$$(1, 1, 1) \to (1, 0, 1) \to (0, 0, 1) \to (0, 1, 1) \to (1, 1, 1).$$

9.4　注释与参考

在 20 世纪 60 年代初, 法国生物学家 Jacob 和 Monod 发现细胞中的调节基因可以打开或关闭其他基因, 从而形成基因回路. 他们因此在 1965 年获得了诺贝尔生物学或医学奖. 在他们发现的启发下, 美国学者 Kauffman 提出用布尔网络来刻画细胞与基因调控网络. 此后, 相关研究取得很大成功, 从而使布尔网络成为系统生理学研究的一个有效工具[89, 142].

早期讨论布尔网络不动点与极限环的文章不少, 但基本上是一篇文章讨论一个系统, 缺少一般性结论. 本章给出的一般公式基于文献 [53]. 除不动点与极限环外, 吸引域 (basin of attraction) 与过渡时间 (transient time) 也是布尔网络拓扑结构的重要组成部分. 详细讨论可见文献 [53].

第 10 章　代数状态空间方法

10.1　状态空间与子空间

为方便计, 本章只考虑 k 值逻辑网络 (即 k 值逻辑动态系统) 的情况. 布尔网络显然是其特殊情况.

考察一个 k 值逻辑动态系统

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t)), \end{cases} \tag{10.1.1}$$

这里, $x_i \in \mathcal{D}_k$, $f_i : \mathcal{D}_k^n \to \mathcal{D}_k$ 为值逻辑函数, $i = 1, \cdots, n$.

或者一个 k 值逻辑控制系统

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \end{cases} \tag{10.1.2}$$
$$y_j(t) = h_j(x_1(t), \cdots, x_n(t)), \quad j = 1, \cdots, p;\ x_i, u_i, y_j \in \mathcal{D}_k.$$

与物理学或控制论中研究的连续状态的力学系统不同, 这里没有自然的状态空间结构. 于是, 我们必须给出一个合理的、便于应用的定义.

定义 10.1.1　考虑 k 值逻辑动态系统 (10.1.1) 或 k 值逻辑控制系统 (10.1.2).

1) 其状态空间, 记作 $\mathcal{X}$, 定义为由 $x_1, \cdots, x_n$ 的所有 k 值逻辑函数组成的集合, 记作 $\mathcal{F}_\ell\{x_1, \cdots, x_n\}$. 即

$$\mathcal{X} = \mathcal{F}_\ell\{x_1, \cdots, x_n\}. \tag{10.1.3}$$

2) 设 $z_1, \cdots, z_r \in \mathcal{X}$. 那么, 由 $z_1, \cdots, z_r$ 生成的子空间, 记作 $\mathcal{Z}$, 指 $z_1, \cdots, z_r$ 的所有 k 值逻辑函数集合. 即

$$\mathcal{Z} = \mathcal{F}_\ell\{z_1, \cdots, z_r\}. \tag{10.1.4}$$

设 $\xi \in \mathcal{X}$. 则 ξ 为 $x_1, \cdots, x_n$ 的一个 k 值逻辑函数. 记

$$\xi = g(x_1, \cdots, x_n).$$

于是, ξ 的向量形式为

$$\xi = M_g \ltimes_{i=1}^n x_i,$$

这里 $M_g \in \mathcal{L}_{k \times k^n}$ 是函数 g 的结构矩阵. 注意到 M_g 可表示为

$$\delta_k[i_1 \; i_2 \; \cdots \; i_{k^n}],$$

这里 $i_s \in \mathcal{D}_k$, $s = 1, \cdots, k^n$. 因此, 可以有 k^{k^n} 个不同的 k 值逻辑函数, 从而有

$$|\mathcal{X}| = k^{k^n}.$$

利用函数集合表示子空间是一种常用的方法. 例如, 在线性代数中, n 维空间上所有的线性函数也是一个 n 维空间, 称为对偶空间. 在泛函分析中: L_p 空间的对偶空间 (原空间的线性泛函空间) 为 $L_q \left(这里, \frac{1}{p} + \frac{1}{q} = 1 \right)$, 等. 用逻辑函数集合表示逻辑动态系统的状态空间及子空间, 其重要原因是: 在向量表示下, 子空间可以与逻辑矩阵相对应. 下面详述这种表示.

设 $\mathcal{Z} = \mathcal{F}_\ell\{z_1, \cdots, z_r\}$ 为 $\mathcal{X} = \mathcal{F}_\ell\{x_1, \cdots, x_n\}$ 的一个子空间, 那么, 每个 $z_i \in \mathcal{X}$ 都是 $x_1, \cdots, x_n$ 的一个 k 值逻辑函数. 于是有

$$z_i = g_i(x_1, \cdots, x_n), \quad i = 1, \cdots, r.$$

设 g_i 的结构矩阵为 M_i, 那么, 在向量形式下有

$$z_i = M_i \ltimes_{j=1}^n x_j, \quad i = 1, \cdots, r.$$

记 $z = \ltimes_{j=1}^r z_j$, $x = \ltimes_{j=1}^n x_j$, 于是有

$$z = M_{\mathcal{Z}} x, \tag{10.1.5}$$

这里

$$M_{\mathcal{Z}} := M_1 * \cdots * M_r \in \mathcal{L}_{k^r \times k^n},$$

称为子空间 $\mathcal{Z}$ 的结构矩阵. 容易证明, 子空间 $\mathcal{Z}$ 的性质完全由其结构矩阵决定.

10.2 状态空间的坐标变换

从现代控制理论中不难看出, 状态空间方法的强大生命力来自坐标变换. 因为适当的坐标能突显相应的各种子空间, 它们刻画了系统不同的特殊性质. 因此, 对于 k 值逻辑动态系统状态空间的坐标变换同样十分重要.

定义 10.2.1 设 $Z=\{z_1,\cdots,z_n\}\subset\mathcal{X}$. 为了记号的方便, 也将 $Z=(z_1,\cdots,z_n)^{\mathrm{T}}$ 当作一个列向量. 由 $X=(x_1,\cdots,x_n)^{\mathrm{T}}\mapsto Z=(z_1,\cdots,z_n)^{\mathrm{T}}$ 所定义的映射 $G:\mathcal{D}_k^n\to\mathcal{D}_k^n$ 称为一个坐标变换, 如果 G 是一一映上的.

下面考虑 G 的具体逻辑表达式:

$$G:\begin{cases}z_1=g_1(x_1,\cdots,x_n),\\ \quad\vdots\\ z_n=g_n(x_1,\cdots,x_n).\end{cases} \tag{10.2.1}$$

利用逻辑的向量表示, 记 $x=\ltimes_{i=1}^n x_i$, $z=\ltimes_{i=1}^n z_i$, 并设 G 的结构矩阵为 M_G. 于是可得到 G 的代数表达式

$$z=M_Gx, \tag{10.2.2}$$

这里 $M_G\in\mathcal{L}_{k^n\times k^n}$. 于是, 显然有如下结论:

定理 10.2.1 G 是一个坐标变换, 当且仅当, 其结构矩阵 M_G 非奇异.

注 如果 $T\in\mathcal{L}_{s\times s}$ 并且非奇异, 则 T 为一正交矩阵. 因此, 如果 G 是一个逻辑坐标变换, 那么, M_G 就是一个正交矩阵, 于是 (10.2.2) 等价于

$$x=M_G^{\mathrm{T}}z. \tag{10.2.3}$$

下面给一个坐标变换的具体例子.

例 10.2.1 令 $k=2$, 即本例中以下均为布尔函数. 设

$$\begin{cases}z_1=\neg x_2,\\ z_2=x_1\leftrightarrow x_2,\\ z_3=\neg x_3.\end{cases} \tag{10.2.4}$$

记 $x=x_1x_2x_3$, $z=z_1z_2z_3$. 那么

$$\begin{aligned}z&=z_1z_2z_3\\ &=M_nx_2M_ex_1x_2M_nx_3\\ &=M_n(I_2\otimes M_e)W_{[2]}x_1x_2^2M_nx_3\\ &=M_n(I_2\otimes M_e)W_{[2]}(I_2\otimes M_{r,2})x_1x_2M_nx_3\\ &=M_n(I_2\otimes M_e)W_{[2]}(I_2\otimes M_{r,2})(I_4\otimes M_n)x_1x_2x_3\\ &:=Tx,\end{aligned} \tag{10.2.5}$$

这里 $T \in \mathcal{L}_{8\times 8}$ 为

$$\begin{aligned} T &= M_n(I_2 \otimes M_e)W_{[2]}(I_2 \otimes M_{r,2})(I_4 \otimes M_n) \\ &= \delta_8[6,5,4,3,8,7,2,1]. \end{aligned} \tag{10.2.6}$$

容易检验 T 是非奇异的, 因此, (10.2.4) 是一个逻辑坐标变换. 其逆变换为

$$\begin{cases} x_1 = z_1 \bar{\vee} z_2, \\ x_2 = \neg z_1, \\ x_3 = \neg z_3. \end{cases} \tag{10.2.7}$$

下面考虑 k 值逻辑动态系统状态空间的逻辑坐标变换.

给定一个 k 值逻辑动态系统, 设其代数状态空间表达式为

$$x(t+1) = Lx(t), \quad x \in \Delta_{k^n}. \tag{10.2.8}$$

令 $z = Tx : \Delta_{k^n} \to \Delta_{k^n}$ 为一 k 值逻辑坐标变换. 那么

$$z(t+1) = Tx(t+1) = TLx(t) = TLT^{-1}z(t).$$

即, 在 z 坐标下逻辑动态系统的动态方程 (10.2.8) 变为

$$z(t+1) = \tilde{L}z(t), \tag{10.2.9}$$

这里

$$\tilde{L} = TLT^{\mathrm{T}}. \tag{10.2.10}$$

下面考虑一个 k 值逻辑控制系统, 设其代数状态空间表达式为

$$\begin{cases} x(t+1) = Lu(t)x(t), & x \in \Delta_{k^n},\ u \in \Delta_{k^m}, \\ y(t) = Hx(t), & y \in \Delta_{k^p}. \end{cases} \tag{10.2.11}$$

设 $z = Tx : \Delta_{k^n} \to \Delta_{k^n}$ 为一逻辑坐标变换. 类似于 k 值逻辑动态系统, k 值逻辑控制系统的动态方程 (10.2.11) 在新坐标下可表示为

$$\begin{aligned} z(t+1) &= \tilde{L}u(t)z(t), && z \in \Delta_{k^n},\ u \in \Delta_{k^m}, \\ y(t) &= \tilde{H}z(t), && y \in \Delta_{k^p}, \end{aligned} \tag{10.2.12}$$

这里

$$\begin{aligned} \tilde{L} &= TL(I_{k^m} \otimes T^{\mathrm{T}}), \\ \tilde{H} &= HT^{\mathrm{T}}. \end{aligned} \tag{10.2.13}$$

下面给一个例子.

例 10.2.2 考虑如下布尔控制网络

$$\begin{cases} x_1(t+1) = x_1(t) \to x_3(t), \\ x_2(t+1) = x_2(t) \leftrightarrow x_3(t), \\ x_3(t+1) = u(t) \wedge x_1(t), \end{cases} \quad (10.2.14)$$
$$y(t) = x_1(t) \leftrightarrow x_2(t).$$

在代数形式下有

$$\begin{aligned} x_1(t+1) &= M_i x_1 x_3(t) &&= M_i F_{[2,2]} x(t) \\ &= M_i F_{[2,2]} R_{[2,8]} u(t) x(t) &&:= M_1 u(t) x(t), \\ x_2(t+1) &= M_e x_2(t) x_3(t) &&= M_e R_{[2,4]} x(t) \\ &= M_e R_{[2,4]} R_{[2,8]} u(t) x(t) &&:= M_2 u(t) x(t), \\ x_3(t+1) &= M_c u(t) x_1(t) &&= M_c u(t) F_{[2,4]} x(t) \\ &= M_c \left(I_2 \otimes F_{[2,4]}\right) u(t) x(t) &&:= M_3 x(t), \end{aligned}$$

这里

$$\begin{aligned} M_1 &= \delta_2[1,2,1,2,1,1,1,1,1,2,1,2,1,1,1,1], \\ M_2 &= \delta_2[1,2,2,1,1,2,2,1,1,2,2,1,1,2,2,1], \\ M_3 &= \delta_2[1,1,1,1,2,2,2,2,2,2,2,2,2,2,2,2]. \end{aligned}$$
$$y(t) = M_e x_1(t) x_2(t) = M_e F_{[4,2]} x(t).$$

于是有

$$\begin{cases} x(t+1) = L u(t) x(t), \\ y(t) = H x(t), \end{cases} \quad (10.2.15)$$

这里

$$\begin{aligned} L &= M_1 * M_2 * M_3 \\ &= \delta_8[1,7,3,5,2,4,4,2,2,8,4,6,2,4,4,2], \\ H &= M_e F_{[4,2]} \\ &= \delta_2[1,1,2,2,2,2,1,1]. \end{aligned}$$

下面考虑坐标变换 (10.2.4). 即 $z = Tx$, 这里

$$T = \delta_8[6,5,4,3,8,7,2,1].$$

利用 (10.2.13), 在 z 坐标下 (10.2.15) 变为

$$\begin{cases} z(t+1) = \tilde{L} u(t) z(t), \\ y(t) = \tilde{H} x(t), \end{cases} \quad (10.2.16)$$

这里

$$
\begin{aligned}
\tilde{L} &= TL(I_2 \otimes T^{\mathrm{T}}) \\
&= \delta_8[5,3,8,4,2,6,3,5,5,3,7,3,1,5,3,5], \\
\tilde{H} &= HT^{\mathrm{T}} \\
&= \delta_2[1,1,2,2,1,1,2,2].
\end{aligned}
$$

10.3　正规子空间

定义 10.3.1　设

$$\mathcal{Z} = \mathcal{F}_\ell\{z_1, \cdots, z_k\} \subset \mathcal{X} = \mathcal{F}_\ell\{x_1, \cdots, x_n\}$$

为一逻辑子空间. $\mathcal{Z}$ 称为一个正规子空间, 如果存在 $\{z_{k+1}, \cdots, z_n\} \subset \mathcal{X}$ 使得

$$T : (x_1, \cdots, x_n) \mapsto (z_1, \cdots, z_n)$$

为一坐标变换.

下面讨论几个例子.

例 10.3.1　给定三变量 (也称三维) 逻辑状态空间 $\mathcal{X} = \mathcal{F}_\ell\{x_1, x_2, x_3\}$. 其子空间 $\mathcal{Z} = \mathcal{F}_\ell\{z_1, z_2\} \subset \mathcal{X}$.

1) 设

$$
\begin{cases}
z_1 = x_1 \leftrightarrow x_2, \\
z_2 = x_2 \bar{\vee} x_3.
\end{cases}
\tag{10.3.1}
$$

容易算出, 它们的代数表达式为

$$
\begin{cases}
z_1 = \delta_2[1\ 1\ 2\ 2\ 2\ 2\ 1\ 1]x_1x_2x_3, \\
z_2 = \delta_2[2\ 1\ 1\ 2\ 2\ 1\ 1\ 2]x_1x_2x_3.
\end{cases}
\tag{10.3.2}
$$

我们通过构造坐标变换来证明 $\mathcal{Z}$ 为一个正规子空间. 为此设

$$z_3 = (x_1 \wedge (x_2 \leftrightarrow x_3)) \vee (\neg x_1 \wedge (x_2 \bar{\vee} x_3)),$$

在向量形式下, 它可表示为

$$z_3 = \delta_2[1\ 2\ 2\ 1\ 2\ 1\ 1\ 2]x_1x_2x_3. \tag{10.3.3}$$

记 $z = z_1z_2z_3$ 及 $x = x_1x_2x_3$, 容易算得

$$z = Lx = \delta_8[3\ 2\ 6\ 7\ 8\ 5\ 1\ 4]x. \tag{10.3.4}$$

因为 $L \in \mathcal{L}_{8\times 8}$ 非奇异, z 是一组新坐标, 因此 $\mathcal{Z}$ 是一个正规子空间.

2) 设

$$\begin{cases} z_1 = x_1 \to x_2, \\ z_2 = x_2 \bar{\vee} x_3. \end{cases} \tag{10.3.5}$$

则它们的代数表达式为

$$\begin{cases} z_1 = \delta_2[1\ 1\ 2\ 2\ 1\ 1\ 1\ 1]x := M_1 x, \\ z_2 = \delta_2[2\ 1\ 1\ 2\ 2\ 1\ 1\ 2]x := M_2 x. \end{cases} \tag{10.3.6}$$

设 $z_3 \in \mathcal{X}$, 其的代数表达式为

$$z_3 = \delta_2[r_1\ r_2\ r_3\ r_4\ r_5\ r_6\ r_7\ r_8] := M_3 x,$$

于是有 $z = Tx$, 这里 $T = M_1 * M_2 * M_3$. 即

$$\mathrm{Col}_i(T) = \mathrm{Col}_i(M_1) \ltimes \mathrm{Col}_i(M_2) \ltimes \mathrm{Col}_i(M_3), \quad i = 1, \cdots, 8.$$

现在

$$\begin{aligned} \mathrm{Col}_2(\mathrm{T}) &= \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_2}, \\ \mathrm{Col}_6(\mathrm{T}) &= \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_6}, \\ \mathrm{Col}_7(\mathrm{T}) &= \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_7}. \end{aligned}$$

因为 $r_i \in \{1, 2\}$, r_2, r_6, r_7 中必有两个相等. 故 T 是奇异的, 即 $\mathcal{Z}$ 不是一个正规子空间.

给定一组逻辑函数 z_i, $i = 1, \cdots, k$, 设其逻辑表达式为

$$z_i = g_i(x_1, \cdots, x_n), \quad i = 1, \cdots, k, \tag{10.3.7}$$

设子空间 $\mathcal{Z} = \mathcal{F}_\ell\{z_1, \cdots, z_r\}$. 我们想知道, 何时 $\mathcal{Z}$ 为一个正规子空间? 如果 $\mathcal{Z}$ 为一个 r 维正规子空间, 则 $\{z_1, \cdots, z_r\}$ 称为它的基底. 设 $\mathcal{Z}$ 的结构矩阵为 $M_\mathcal{Z}$, 即

$$z = M_\mathcal{Z} x,$$

这里 $M_\mathcal{Z} \in \mathcal{L}_{k^r \times k^n}$. 从前面的例子不难得到如下的一般性结论.

定理 10.3.1 设 $\mathcal{Z} = \mathcal{F}_\ell\{z_1, \cdots, z_r\}$. $\mathcal{Z}$ 为一个 r 维正规子空间, 且 $\{z_1, \cdots, z_r\}$ 为其基底, 当且仅当, $M_\mathcal{Z}$ 具有 k^r 组互不相同的列, 而每一组的个数均为 k^{n-r}.

实际上, 也可以直接由 g_i, $i = 1, \cdots, k$ 的结构矩阵来判定正规子空间. 设 g_i 的结构矩阵为

$$M_i = \delta_k[r_{i,1}, \cdots, r_{i,k^n}], \quad i = 1, \cdots, r.$$

定义其示性矩阵为

$$G(M_1,\cdots,M_k):=\begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,k^n} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,k^n} \\ \vdots & \vdots & & \vdots \\ r_{k,1} & r_{k,2} & \cdots & r_{k,k^n} \end{bmatrix}. \tag{10.3.8}$$

那么

推论 10.3.1 $\mathcal{Z}$ 为一个 r 维正规子空间, 且 $\{z_1,\cdots,z_r\}$ 为其基底, 当且仅当, G 具有 k^r 组互不相同的列, 而每一组的个数均为 k^{n-r}.

下面通过几个例子说明怎样从正规子空间构造新坐标.

例 10.3.2 1) 回忆例 10.3.1 的第一个例子. 由式 (10.3.2) 可得

$$G=\begin{bmatrix} 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 \end{bmatrix}.$$

它显然满足推论 10.3.1 的要求. 要构造新坐标, 只要补上的坐标变量使扩充的 G 的各列都不相同即可. 例如, [1 2 2 1 2 1 1 2], 显然满足要求. 其实, 这就是 (10.3.3). 还有许多选择, 对这个例子, 只要 $r_1\neq r_8, r_2\neq r_7, r_3\neq r_6, r_4\neq r_5$ 就可以了. 所以, 可以选 $z_3=\delta_2[r_1,r_2,r_3,r_4,3-r_4,3-r_3,3-r_2,3-r_1]$, 这里 $r_1,r_2,r_3,r_4\in\{1,2\}$ 可任选. 例如 $z_3=\delta_2[2,1,1,2,1,2,2,1]$ 等.

2) 设 $\mathcal{X}=\mathcal{F}_\ell\{x_1,x_2,x_3,x_4\}$, $\mathcal{Z}=\mathcal{F}_\ell\{z_1,z_2\}$, 这里

$$\begin{aligned} z_1&=\delta_2[2\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 2\ 2\ 1\ 1\ 2\ 1\ 2\ 2]x,\\ z_2&=\delta_2[2\ 1\ 1\ 2\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 1\ 2\ 2\ 1\ 1]x. \end{aligned}$$

容易看出, $\mathcal{Z}$ 是一个正规子空间. 要利用 $\{z_1,z_2\}$ 构造新坐标, 只要选择 $\{z_3,z_4\}$ 使示性矩阵 $G(z_1,z_2,z_3,z_4)$ 的各列均不相同即可. 以下为一种选法:

$$\begin{aligned} z_3&=\delta_2[1\ 1\ 2\ 2\ 1\ 2\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 1\ 1\ 2]x,\\ z_4&=\delta_2[1\ 2\ 2\ 1\ 1\ 2\ 1\ 1\ 2\ 2\ 2\ 1\ 1\ 2\ 2\ 1]x. \end{aligned}$$

在结束本节前, 我们讨论一下对于布尔函数, 如何从逻辑变量的代数表达式返回逻辑表示式.

命题 10.3.1 设 $z=z(x_1,\cdots,x_n)$ 为一逻辑函数, 其结构矩阵为 M_z, 即

$$z=M_z\ltimes_{i=1}^n x_i, \tag{10.3.9}$$

这里 $M_z\in\mathcal{L}_{2\times 2^n}$. 记 $M_z=[M_1,M_2]$, 这里 $M_i\in\mathcal{L}_{2\times 2^{n-1}}$, $i=1,2$. 那么

$$z=[x_1\wedge f_1(x_2,\cdots,x_n)]\vee[\neg x_1\wedge f_2(x_2,\cdots,x_n)], \tag{10.3.10}$$

这里 f_i 的结构矩阵为 M_i, $i=1,2$.

下面介绍几个记号:

(i) 设 $0 \leqslant i \leqslant 2^n - 1$, $\mu^n(i) := [\mu_1^n(i), \mu_2^n(i) \cdots, \mu_n^n(i)]$ 为 i 的二进制表示, 即 $\mu_k^n(i) \in \{0, 1\}$, $k = 1, \cdots, n$, 并且

$$\sum_{k=1}^{n} \mu_k^n(i) \times 2^{n-k} = i.$$

(ii) 记 $\neg^0$ 为恒等算子, 即 $\neg^0 x \equiv x$, $x \in \{0, 1\}$. $\neg^1 = \neg$, 即 $\neg^1 x \equiv \neg x$.

(iii) 记 $z = z(x_1, \cdots, x_n)$ 的结构矩阵为

$$M_z = [M_1, M_2, \cdots, M_{2^{n-1}}], \tag{10.3.11}$$

这里 $M_i \in \mathcal{L}_{2\times 2}$, $i = 1, \cdots, 2^{n-1}$.

利用上述记号, 如果反复应用命题 10.3.1, 即可得到以下的结果:

推论 10.3.2 设 $z = z(x_1, \cdots, x_n)$ 的结构矩阵为 (10.3.11), 则

$$\begin{aligned} z = \bigvee_{i=1}^{2^{n-1}} & \left(\neg^{\mu_1^{n-1}}(i-1)x_1\right) \wedge \left(\neg^{\mu_2^{n-1}}(i-1)x_2\right) \\ & \wedge \cdots \wedge \left(\neg^{\mu_{n-1}^{n-1}}(i-1)x_{n-1}\right) \wedge f_i(x_n), \end{aligned} \tag{10.3.12}$$

这里 f_i 的结构矩阵为 M_i, $i = 1, \cdots, 2^{n-1}$.

注意, M_i 有四种可能形态, 它们分别导出如下的逻辑表达式:

$$M_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow f_i(x_n) = x_n,$$

$$M_i = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \Rightarrow f_i(x_n) = \neg x_n,$$

$$M_i = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \Rightarrow f_i(x_n) = 1,$$

$$M_i = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} \Rightarrow f_i(x_n) = 0.$$

下面计算一个例子.

例 10.3.3 在例 10.3.2 的 2) 中,

$$z_1 = \delta_2[2\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 2\ 2\ 1\ 1\ 2\ 1\ 2\ 2]x.$$

可得

$$
\begin{aligned}
&M_1 = M_4 = M_7 = \delta_2[2\ 1] \Rightarrow f_1(x_4) = f_4(x_4) = f_7(x_4) = \neg(x_4),\\
&M_2 = M_6 = \delta_2[1\ 1] \quad\ \Rightarrow f_2(x_4) = f_6(x_4) = 1,\\
&M_3 = \delta_2[1\ 2] \qquad\qquad\ \ \Rightarrow f_3(x_4) = x_4,\\
&M_5 = M_8 = \delta_2[2\ 2] \quad\ \Rightarrow f_5(x_4) = f_8(x_4) = 0.
\end{aligned}
$$

最后可得

$$
\begin{aligned}
z_1 = &(x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4) \vee (x_1 \wedge x_2 \wedge x_3 \wedge 1)\\
&\vee (x_1 \wedge \neg x_2 \wedge x_3 \wedge x_4) \vee (x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge \neg x_4)\\
&\vee (\neg x_1 \wedge x_2 \wedge x_3 \wedge 0) \vee (\neg x_1 \wedge x_2 \wedge x_3 \wedge 1)\\
&\vee (\neg x_1 \wedge \neg x_2 \wedge x_3 \wedge \neg x_4) \vee (\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge 0)\\
= &(x_1 \wedge x_2 \wedge x_3 \wedge \neg x_4) \vee (x_1 \wedge x_2 \wedge x_3)\\
&\vee (x_1 \wedge \neg x_2 \wedge x_3 \wedge x_4) \vee (x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge \neg x_4)\\
&\vee (\neg x_1 \wedge x_2 \wedge x_3 \wedge 1) \vee (\neg x_1 \wedge \neg x_2 \wedge x_3 \wedge \neg x_4).
\end{aligned}
$$

z_2, z_3, z_4 的计算留做练习.

10.4 不变子空间

考虑一个 k 值逻辑动态系统

$$
\begin{cases}
x_1(t+1) = f_1(x_1, \cdots, x_n),\\
x_2(t+1) = f_2(x_1, \cdots, x_n),\\
\quad\vdots\\
x_n(t+1) = f_n(x_1, \cdots, x_n).
\end{cases} \tag{10.4.1}
$$

定义 10.4.1 考虑逻辑动态系统 (10.4.1). $\mathcal{Z} = \mathcal{F}_\ell\{z^1\} = \mathcal{F}_\ell\{z_1^1, \cdots, z_s^1\}$ 称为系统的不变子空间, 如果存在 $z^2 = \{z_1^2, \cdots, z_{n-s}^2\}$, 使 $x \mapsto z = \{z^1, z^2\}$ 为一坐标变换, 并且, 在 z 坐标下系统可表示为

$$
\begin{cases}
z^1(t+1) = F^1(z^1(t)), & z^1 \in \mathcal{D}_k^s,\\
z^2(t+1) = F^2(z(t)), & z^2 \in \mathcal{D}_k^{n-s}.
\end{cases} \tag{10.4.2}
$$

设 (10.4.1) 的代数状态空间表示为

$$
x(t+1) = Lx(t). \tag{10.4.3}
$$

则有如下结论:

定理 10.4.1 考查系统 (10.4.1) 及其代数形式 (10.4.3). 设 $\mathcal{Z}=\mathcal{F}_\ell\{z_1,\cdots,z_s\}$ 为正规子空间, 其代数表达式为

$$z=Qx, \tag{10.4.4}$$

这里 $z=\ltimes_{i=1}^s z_i$, $Q\in\mathcal{L}_{k^s\times k^n}$. 那么

1) $\mathcal{Z}$ 为系统 (10.4.1) 的不变子空间, 当且仅当,

$$\mathrm{Row}(QL)\subset \mathrm{span}_{\mathcal{B}}\mathrm{Row}(Q), \tag{10.4.5}$$

这里 $\mathrm{span}_{\mathcal{B}}$ 指系数属于 $\mathcal{D}$.

2) 条件 (10.4.5) 等价于: 存在 $H\in\mathcal{L}_{2^s\times 2^s}$, 使得

$$QL=HQ. \tag{10.4.6}$$

下面给一个例子.

例 10.4.1 考查如下系统

$$\begin{cases}x_1(t+1)=(x_1(t)\wedge x_2(t)\wedge\neg x_4(t))\vee(\neg x_1(t)\wedge x_2(t)),\\ x_2(t+1)=x_2(t)\vee(x_3(t)\leftrightarrow x_4(t)),\\ x_3(t+1)=(x_1(t)\wedge\neg x_4(t))\vee(\neg x_1(t)\wedge x_2(t))\vee(\neg x_1(t)\wedge\neg x_2(t)\wedge x_4(t)),\\ x_4(t+1)=x_1(t)\wedge\neg x_2(t)\wedge x_4(t).\end{cases} \tag{10.4.7}$$

易知, 其代数表示为

$$x(t+1)=Lx(t), \tag{10.4.8}$$

这里

$$L=\delta_{16}[11\ 1\ 11\ 1\ 11\ 13\ 15\ 9\ 1\ 2\ 1\ 2\ 9\ 15\ 13\ 11].$$

设 $\mathcal{Z}=\mathcal{F}_\ell\{z_1,z_2,z_3\}$, 这里

$$\begin{cases}z_1=x_1\bar{\vee}x_4,\\ z_2=\neg x_2,\\ z_3=x_3\leftrightarrow\neg x_4.\end{cases} \tag{10.4.9}$$

记 $x=\ltimes_{i=1}^4 x_i$, $z=\ltimes_{i=1}^3 z_i$. 则有

$$z=Qx,$$

这里

$$Q = \delta_8[8\ 3\ 7\ 4\ 6\ 1\ 5\ 2\ 4\ 7\ 3\ 8\ 2\ 5\ 1\ 6],$$

容易检验

$$QL = \delta_8[3\ 8\ 3\ 8\ 3\ 2\ 1\ 4\ 8\ 3\ 8\ 3\ 4\ 1\ 2\ 3],$$

它满足 (10.4.5). 因此 $\mathcal{Z}$ 是系统 (10.4.7) 的不变子空间.

实际上, 可选 $z_4 = x_4$, 使得

$$\begin{cases} z_1 = x_1 \bar{\vee} x_4, \\ z_2 = \neg x_2, \\ z_3 = x_3 \leftrightarrow \neg x_4, \\ z_4 = x_4. \end{cases} \tag{10.4.10}$$

为一坐标变换. 而在这个新坐标下, 系统 (10.4.7) 变为

$$\begin{cases} z_1(t+1) = z_1(t) \to z_2(t), \\ z_2(t+1) = z_2(t) \wedge z_3(t), \\ z_3(t+1) = \neg z_1(t), \\ z_4(t+1) = z_1(t) \vee z_2(t) \vee z_4(t). \end{cases} \tag{10.4.11}$$

(10.2.12) 显然具有 (10.4.2) 的标准形式.

10.5　注释与参考

Kalman 提出的状态空间方法是现代控制理论的三大支柱之一. 它从线性系统开始, 推广到非线性系统, 再到分布参数系统的泛函空间, 成为系统分析与综合的基本平台. 对于逻辑动态系统, 如何定义适当的状态空间, 从而引入有效的状态空间方法, 这对逻辑动态系统的分析与控制极具重要性. 本章内容主要来自文献 [54].

(A, B) 不变子空间 (也称能控不变子空间) 在线性系统分析与控制设计中起重要作用. 对逻辑控制系统 (或特殊的布尔控制系统) 尚未见到对相应能控不变子空间的讨论.

第 11 章　逻辑控制系统的能控性与能观性

11.1　可达与能控性

本章只考虑 k 值逻辑系统的能控性与能观性. 考察一个 k 值逻辑控制系统

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \end{cases}$$
$$y_j(t) = h_j(x_1(t), \cdots, x_n(t)), \quad j = 1, \cdots, p, \tag{11.1.1}$$

这里 $x_i, u_i, y_j \in \mathcal{D}_k$.

记其代数状态空间表示为

$$\begin{aligned} x(t+1) &= Lu(t)x(t), \\ y(t) &= Hx(t), \end{aligned} \tag{11.1.2}$$

这里, $L \in \mathcal{L}_{k^n \times k^{n+m}}$, $H \in \mathcal{L}_{k^p \times k^n}$.

当我们考虑系统能控性时, 暂时不必考虑输出.

定义 11.1.1　考虑系统 (11.1.1).

(i) 给定初态 $x_0 = (x_0^1, \cdots, x_0^n)$ 及末态 $x_d = (x_d^1, \cdots, x_d^n)$. 称 x_d 自 x_0 可达的, 如果存在 $T > 0$, 控制 $\{u(t), t = 1, \cdots, T-1\}$, 使系统轨线 $x(u, t)$ 在该控制驱动下, 可以从 $x(u, 0) = x_0$ 出发, 在 T 时刻到达 x_d, 即 $x(u, T) = x_d$.

(ii) 称系统是能控的, 如果对任意 x_0 和 x_d, x_d 自 x_0 可达.

注意, 我们使用 x 表示系统变量的两种形式: (i) 分量形式 $x = (x_1, \cdots, x_n)$, $x_i \in \mathcal{D}_k$; (ii) 乘积形式 $x = \ltimes_{i=1}^n x_i$, $x_i \in \Delta_k$. 它们是一一对应的. 读者不难从前后文看出它是哪种形式.

11.1.1　网络输入

本节假设控制是由一控制网络决定的, 即

$$\begin{cases} u_1(t+1) = g_1(u_1(t), \cdots, u_m(t)), \\ \quad\vdots \\ u_m(t+1) = g_m(u_1(t), \cdots, u_m(t)). \end{cases} \tag{11.1.3}$$

其代数状态空间表示为

$$u(t+1) = Gu(t), \tag{11.1.4}$$

这里, $G \in \mathcal{L}_{k^m \times k^m}$.

定义 11.1.2　给定控制网络的结构矩阵 G, 相应的控制-状态传递矩阵, 记作 $\Theta^G(t,0)$, 定义如下: 对任一 $u_0 \in \Delta_{k^m}$ 和 $x_0 \in \Delta_{k^n}$, 有

$$x(t) = \Theta^G(t,0)u_0x_0, \quad x(t) \in \Delta_{k^n},\ t > 0. \tag{11.1.5}$$

利用数学归纳法, 不难证明以下公式:

命题 11.1.1　控制-状态传递矩阵可计算如下:

$$\begin{aligned}\Theta^G(t,0) = {} & LG^{t-1}(I_{k^m} \otimes LG^{t-2})(I_{k^{2m}} \otimes LG^{t-3}) \cdots (I_{k^{(t-1)m}} \otimes L) \\ & (I_{k^{(t-2)m}} \otimes M_{r,k^m})(I_{k^{(t-3)m}} \otimes M_{r,k^m}) \cdots (I_{k^m} \otimes M_{r,k^m})M_{r,k^m}.\end{aligned} \tag{11.1.6}$$

其中, M_{r,k^m} 是降阶矩阵.

利用公式 (11.1.5)–(11.1.6) 即可检验系统可控性:

定理 11.1.1　考虑系统 (11.1.1) 及控制律 (11.1.4). x_d 是自 x_0 出发 s 步可到达的, 当且仅当

$$x_d \in \mathrm{Col}\left\{\Theta^G(s,0)W_{[k^n,k^m]}x_0\right\}. \tag{11.1.7}$$

下面给一个例子.

例 11.1.1　考虑一个布尔控制系统

$$\begin{cases} x_1(t+1) = x_2(t) \leftrightarrow x_3(t), \\ x_2(t+1) = x_3(t) \vee u_1(t), \\ x_3(t+1) = x_1(t) \wedge u_2(t). \end{cases} \tag{11.1.8}$$

设控制网络为

$$\begin{cases} u_1(t+1) = g_1(u_1(t), u_2(t)), \\ u_2(t+1) = g_2(u_1(t), u_2(t)), \end{cases} \tag{11.1.9}$$

其中

$$\begin{cases} g_1(u_1(t), u_2(t)) = \neg u_2(t), \\ g_2(u_1(t), u_2(t)) = u_1(t). \end{cases} \tag{11.1.10}$$

又设 $x_1(0)=1$, $x_2(0)=0$, $x_3(0)=1$, 考虑 $s=5$. 记 $u(t)=u_1(t)u_2(t)$, 则

$$u(t+1)=u_1(t+1)u_2(t+1)=M_nu_2(t)u_1(t)=M_nW_{[2]}u(t).$$

故

$$G=M_nW_{[2]}=\delta_4[3\ 1\ 4\ 2],$$

$$x(t+1)=M_eB(t)C(t)M_dC(t)u_1(t)M_cA(t)u_2(t)=Lx(t),$$

这里 $L\in\mathcal{L}_{8\times 32}$ 为

$$L=\delta_8[1\ 5\ 5\ 1\ 2\ 6\ 6\ 2\ 2\ 6\ 6\ 2\ 2\ 6\ 6\ 2\ 1\ 7\ 5\ 3\ 2\ 8\ 6\ 4\ 2\ 8\ 6\ 4\ 2\ 8\ 6\ 4].$$

$$M_{r,2^2}=\delta_{16}[1\ 6\ 11\ 16].$$

利用公式 (11.1.6), 则 $\Theta(5,0)\in\mathcal{L}_{8\times 32}$ 为

$$\begin{aligned}\Theta(5,0)=&LG^4(I_{2^6}\otimes LG^3)(I_{2^4}\otimes LG^2)(I_{2^6}\otimes LG)(I_{2^8}\otimes L)\\&(I_{2^6}\otimes M_{r,2^2})(I_{2^4}\otimes M_{r,2^2})(I_{2^2}\otimes M_{r,2^2})M_{r,2^2}\\=&\delta_8[6\ 5\ 5\ 6\ 6\ 5\ 5,6\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\\&8\ 8\ 8\ 8\ 2\ 2\ 2\ 2\ 4\ 8\ 4\ 8\ 4\ 8\ 4\ 8].\end{aligned}$$

设 $x_0=(x_1(0),x_2(0),x_3(0))=(1,0,1)$, 即

$$x_0=x_1(0)x_2(0)x_3(0)=\delta_2^1\delta_2^2\delta_2^1=\delta_8^3.$$

于是可得

$$\Theta(5,0)W_{[8,4]}x_0=\delta_8[5\ 2\ 8\ 4].$$

记可达集为 $R_5(x_0)$, 根据定理 11.1.1, 可知

$$R_5((1,0,1))=\mathrm{Col}\left(\Theta(5,0)W_{[8,4]}x_0\right)=\delta_8\left\{5\ 2\ 8\ 4\right\}.$$

转为布尔形式, 有

$$R_5((1,0,1))=\{(0,1,1),(1,1,0),(0,0,0),(1,0,0)\}.$$

最后, 要找出相应的初始控制. 因为

$$x_d=\Theta(5,0)W_{[8,4]}x_0u_0=\delta_8[5\ 2\ 8\ 4]u_0.$$

要达到 $\delta_8^5\sim(0,1,1)$, 则 $u_0=\delta_4^1$, 即 $u_1(0)=1$, $u_2(0)=1$.

类似地, 要达到其他 3 点:

$$\{(1,1,0);\ (0,0,0);\ (1,0,0)\},$$

相应的控制为

$$(u_1(0),u_2(0))=\{(1,0);(0,1);(0,0)\}.$$

11.1.2 自由输入

考虑系统 (11.1.1). 对给定 $x_0, x_d \in \Delta_{k^n}$, 考察是否存在布尔控制序列 $u(t) \in \mathcal{D}_k^m$, $t = 0, 1, \cdots, T-1$, 使相应轨线从 $x(0) = x_0$ 在该控制下到达 $x(T) = x_d$.

定义 $\tilde{L} = LW_{[2^n, 2^m]}$, 那么, 式 (11.1.2) 的状态方程可表示为

$$x(t+1) = \tilde{L}x(t)u(t). \tag{11.1.11}$$

重复利用这个表达式可得

$$x(T) = \tilde{L}^{\mathrm{T}}x(0)u(0)u(1)\cdots u(T-1). \tag{11.1.12}$$

因此, 不难得到以下结论:

定理 11.1.2 *存在布尔控制序列, 使 x_d 从 x_0 出发 T 步可到达, 当且仅当,*

$$x_d \in \mathrm{Col}\{\tilde{L}^{\mathrm{T}}x_0\}. \tag{11.1.13}$$

考察如下例子.

例 11.1.2 给定一个布尔控制系统

$$\begin{cases} x_1(t+1) = x_3(t) \wedge u_1(t), \\ x_2(t+1) = \neg u_2(t), \\ x_3(t+1) = x_1(t) \vee x_2(t). \end{cases} \tag{11.1.14}$$

其向量形式为

$$\begin{cases} x_1(t+1) = M_c x_3(t) u_1(t), \\ x_2(t+1) = M_n u_2(t), \\ x_3(t+1) = M_d x_1(t) x_2(t). \end{cases} \tag{11.1.15}$$

记 $x(t) = x_1(t)x_2(t)x_3(t)$, $u(t) = u_1(t)u_2(t)$. 容易算出

$$x(t+1) = \tilde{L}x(t)u(t), \tag{11.1.16}$$

这里 $\tilde{L} \in \mathcal{L}_{8\times 32}$ 为

$$\begin{aligned} \tilde{L} = \delta_8[&3\ 1\ 7\ 5\ 3\ 1\ 7\ 5\ 7\ 5\ 7\ 5\ 7\ 5\ 7\ 5 \\ &3\ 1\ 7\ 5\ 3\ 1\ 7\ 5\ 8\ 6\ 8\ 6\ 8\ 6\ 8\ 6]. \end{aligned}$$

设 $x_0=(x_1(0),x_2(0),x_3(0))=(0,0,0)$. 我们考虑 $T=3$ 的可达集. $\tilde{L}^3x_0\in\mathcal{L}_{8\times 64}$ 为

$$
\begin{aligned}
\tilde{L}^3x_0=\delta_8[&8\ 6\ 8\ 6\ 3\ 1\ 7\ 5\ 8\ 6\ 8\ 6\ 3\ 1\ 7\ 5\\
&7\ 5\ 7\ 5\ 3\ 1\ 7\ 5\ 8\ 6\ 8\ 6\ 3\ 1\ 7\ 5\\
&8\ 6\ 8\ 6\ 3\ 1\ 7\ 5\ 8\ 6\ 8\ 6\ 3\ 1\ 7\ 5\\
&7\ 5\ 7\ 5\ 3\ 1\ 7\ 5\ 8\ 6\ 8\ 6\ 3\ 1\ 7\ 5].
\end{aligned}
$$

根据定理 11.1.2, 除 δ_8^2 及 δ_8^4 外, 其他状态均可达. 任选一可达状态, 例如 $\delta_8^5\sim(0,1,1)$. 因为 $\tilde{L}^3x_0$ 的第 8 列, 第 16 列, 第 18 列, 第 20 列等为 δ_8^5, 则控制 δ_{64}^8, 或 δ_{64}^{16}, 或 δ_{64}^{18}, 或 δ_{64}^{20} 等, 可将初态 $(0,0,0)$ 驱至 $(0,1,1)$. 例如选 δ_{64}^8, 即

$$u_1(0)u_2(0)u_1(1)u_2(1)u_1(2)u_2(2)=\delta_{64}^8.$$

不难算出

$$u_1(0)=1;\ u_2(0)=1;\ u_1(1)=1;\ u_2(1)=0;\ u_1(2)=0;\ u_2(2)=0.$$

不难验证, 对该系统, 取 $T=1$, $x_0=(0,0,0)$ 的可达集为

$$R_1(x_0)=\{(0,1,0),(0,0,0)\}.$$

当 $T>1$ 时, 可达集为

$$R_T(x_0)=\{(1,1,1),(1,0,1),(0,1,1),(0,1,0),(0,0,1),(0,0,0)\},\quad T>1.$$

由定理 11.1.2 可推得

推论 11.1.1 x_d 是从 x_0 可达, 当且仅当

$$x_d\in\mathrm{Col}\{\cup_{i=1}^{\infty}\tilde{L}^ix_0\}.\tag{11.1.17}$$

记 $R(x_0)$ 为 x_0 的可达集, 则有如下结果:

命题 11.1.2 1) $R(x_0)$ 是 $\mathrm{Col}\{\tilde{L}\}$ 的子集;

2) 设 r^* 为最小的 $r>0$, 使得

$$\mathrm{Col}\{\tilde{L}^{r+1}x_0\}\subset\mathrm{Col}\{\tilde{L}^sx_0\,\big|\,s=1,2,\cdots,r\},$$

那么, 可达集

$$R(x_0)=\mathrm{Col}\{\cup_{i=1}^{r^*}\tilde{L}^ix_0\}.\tag{11.1.18}$$

11.2 能 观 性

定义 11.2.1　考察逻辑控制系统 (11.1.1).

1) 状态 x_1 和 x_2 称为可区分的, 如果存在 $T \geqslant 0$, 控制序列 $\{u(0), u(1), \cdots, u(T)\}$, 使得初值分别为 $x_0 = x_1$ 与 $x_0 = x_2$ 的两条轨线在 T 时刻不等, 即

$$\begin{aligned} y^1(T) &= y(T)(u(T-1), \cdots, u(0), x_1) \\ &\neq y^2(T) = y(T)(u(T-1), \cdots, u(0), x_2). \end{aligned} \tag{11.2.1}$$

2) 系统 (11.1.1) 称为能观的, 如果任何两个初值点 $x_0, y_0 \in \mathcal{D}_{2^n}$ 均可区分.

我们通过如下步骤构造一个能观性矩阵:

- 步骤 1. 依如下步骤构造一个矩阵序列 $\Gamma_i \in \mathcal{L}_{2^p \times 2^n}$, $i = 1, 2, \cdots$:

$$\begin{aligned} \Gamma_1 &= \left\{ L\delta_{k^m}^i \,|\, i = 1, 2, \cdots, 2^m \right\}; \\ \Gamma_{r+1} &= \left\{ L\delta_{k^m}^i \gamma \,|\, \gamma \in \Gamma_r;\ i = 1, 2, \cdots, 2^m \right\}, \quad r \geqslant 1. \end{aligned}$$

如果 $\mathrm{Col}\{\Gamma_{r^*+1}\} \subset \mathrm{Col}\{\Gamma_i \,|\, i \leqslant r^*\}$, 则 $r^* + 1$ 称为退化步骤. 设 $r^* > 0$ 为最后一个非退化步骤, 序列止步于 r^*. (由于序列元素至多有 k^n 个不同的列, $r^* \leqslant k^n$.)

- 步骤 2. 构造另一组序列 $H_i \in \mathcal{L}_{k^p \times k^n}$: 这里 $H_0 = H$,

$$H_i = H\Gamma_i = \{H\gamma \,|\, \gamma \in \Gamma_i\}, \quad i \leqslant r^*.$$

- 步骤 3. 设 $h_j^i = \delta_{k^p}[s_1, \cdots, s_{2^n}] \in H_i$, 简化其为

$$[h]_j^i = (s_1, \cdots, s_{2^n}).$$

选择 H_i 中线性无关向量:

$$[h]_j^i \in H_i, \quad j = 1, \cdots, n_i;\ i = 1, \cdots, r^*,$$

构造能观矩阵如下:

$$\mathcal{C} = \begin{bmatrix} [h]^0 \\ [h]_1^1 \\ \vdots \\ [h]_{n_1}^1 \\ \vdots \\ [h]_1^{r^*} \\ \vdots \\ [h]_{n_{r^*}}^{r^*} \end{bmatrix}. \tag{11.2.2}$$

称 $\mathcal{C}$ 为能观性矩阵.

定理 11.2.1 设系统 (11.1.1) 全局能控. 则系统能观, 当且仅当, $\mathcal{C}$ 的任意两列均不相同.

下面的例子来自文献 [55].

例 11.2.1 考查一个布尔控制系统

$$\begin{cases} x_1(t+1) = x_2(t) \leftrightarrow x_3(t), \\ x_2(t+1) = x_3(t) \vee u_1(t), \\ x_3(t+1) = x_1(t) \wedge u_2(t), \end{cases} \\ y_1(t) = x_1(t), \\ y_2(t) = x_2(t) \vee x_3(t). \tag{11.2.3}$$

易知, 其代数表达式为

$$\begin{aligned} x(t+1) &= Lx(t), \\ y(t) &= Hx(t), \end{aligned} \tag{11.2.4}$$

这里

$$\begin{aligned} L = \delta_8[&1\ 5\ 5\ 1\ 2\ 6\ 6\ 2\ 2\ 6\ 6\ 2\ 2\ 6\ 6\ 2 \\ &1\ 7\ 5\ 3\ 2\ 8\ 6\ 4\ 2\ 8\ 6\ 4\ 2\ 8\ 6\ 4], \\ H = \delta_4[&1\ 1\ 1\ 2\ 3\ 3\ 3\ 4]. \end{aligned}$$

直接计算可得

$$\begin{aligned} HL\delta_4^1 &= \delta_4[1\ 3\ 3\ 1\ 1\ 3\ 3\ 1]; \\ HL\delta_4^2 &= \delta_4[1\ 3\ 3\ 1\ 1\ 3\ 3\ 1]; \\ HL\delta_4^3 &= \delta_4[1\ 3\ 3\ 1\ 1\ 4\ 3\ 2]; \\ HL\delta_4^4 &= \delta_4[1\ 4\ 3\ 2\ 1\ 4\ 3\ 2]. \end{aligned}$$

于是, 能观性矩阵为

$$\mathcal{C} = \begin{bmatrix} H \\ HL\delta_4^1 \\ HL\delta_4^2 \\ HL\delta_4^3 \\ HL\delta_4^4 \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 2 & 3 & 3 & 3 & 4 \\ 1 & 3 & 3 & 1 & 1 & 3 & 3 & 1 \\ 1 & 3 & 3 & 1 & 1 & 4 & 3 & 2 \\ 1 & 4 & 3 & 2 & 1 & 4 & 3 & 2 \\ & & & \vdots & & & & \end{bmatrix}.$$

显然, $\mathcal{C}$ 没有相同的列. 容易检验, 该系统全局能控[55]. 因此, 由定理 11.2.1 可知, 该系统能观.

11.3 输入–状态关联矩阵

考虑一个 k 值逻辑控制系统 (11.1.1). 如果对 $x(t)=x$ 存在控制 u 使 $x(t+1)=y$, 则称 y 是 (u,x) 可达的, 因为下一个时刻的控制 $u(t+1)$ 可以任选, 我们也可以说, 对任意 $v\in\Delta_{k^m}$, (v,y) 是 (u,x) 可达的, 记作 $(u,x)\to(v,y)$. 这样, 我们可以在输入–状态乘积空间中定义一个输入–状态转移图. 它的顶点是

$$V=\{(u,x)\mid u\in\mathcal{D}_k^m, x\in\mathcal{D}_k^n\},$$

边是

$$E=\{[(u,x),(v,y)]\mid(u,x)\to(v,y)\}.$$

定义 11.3.1 一个 k 值逻辑控制系统的输入–状态转移图的邻接矩阵称为该系统的输入–状态关联矩阵, 记为 $\mathcal{J}$.

我们用下面的例子来说明它.

例 11.3.1 考虑下列布尔控制系统

$$\begin{cases}x_1(t+1)=(x_1(t)\vee x_2(t))\wedge u(t),\\ x_2(t+1)=x_1(t)\leftrightarrow u(t).\end{cases}\tag{11.3.1}$$

记 $x(t)=x_1(t)\ltimes x_2(t)$, 容易算出其代数表达式为

$$x(t+1)=Lu(t)x(t),\tag{11.3.2}$$

这里

$$L=\delta_4[1\ 1\ 2\ 4\ 4\ 4\ 3\ 3].\tag{11.3.3}$$

简记 $(u,(x_1,x_2))\sim(u,x_1,x_2)$, 则输入–状态转移图的顶点为

$$V=\{(111),(110),(101),(100),(011),(010),(001),(000)\}:=\{V_i|i=1,2,\cdots,8\}.$$

现在, 如果 $u(t)=1$, $x_1(t)=1$, $x_2(t)=1$, 则由系统 (11.3.1) 可得 $x_1(t+1)=1$, $x_2(t+1)=1$. 根据定义, 有

$$[(1,1,1),(1,1,1)]\in E,\quad [(1,1,1),(0,1,1)]\in E.$$

即 $[V_1,V_1]\in E$, $[V_1,V_5]\in E$.

类似地可得所有的边. 于是有系统的输入–状态转移图如图 11.3.1.

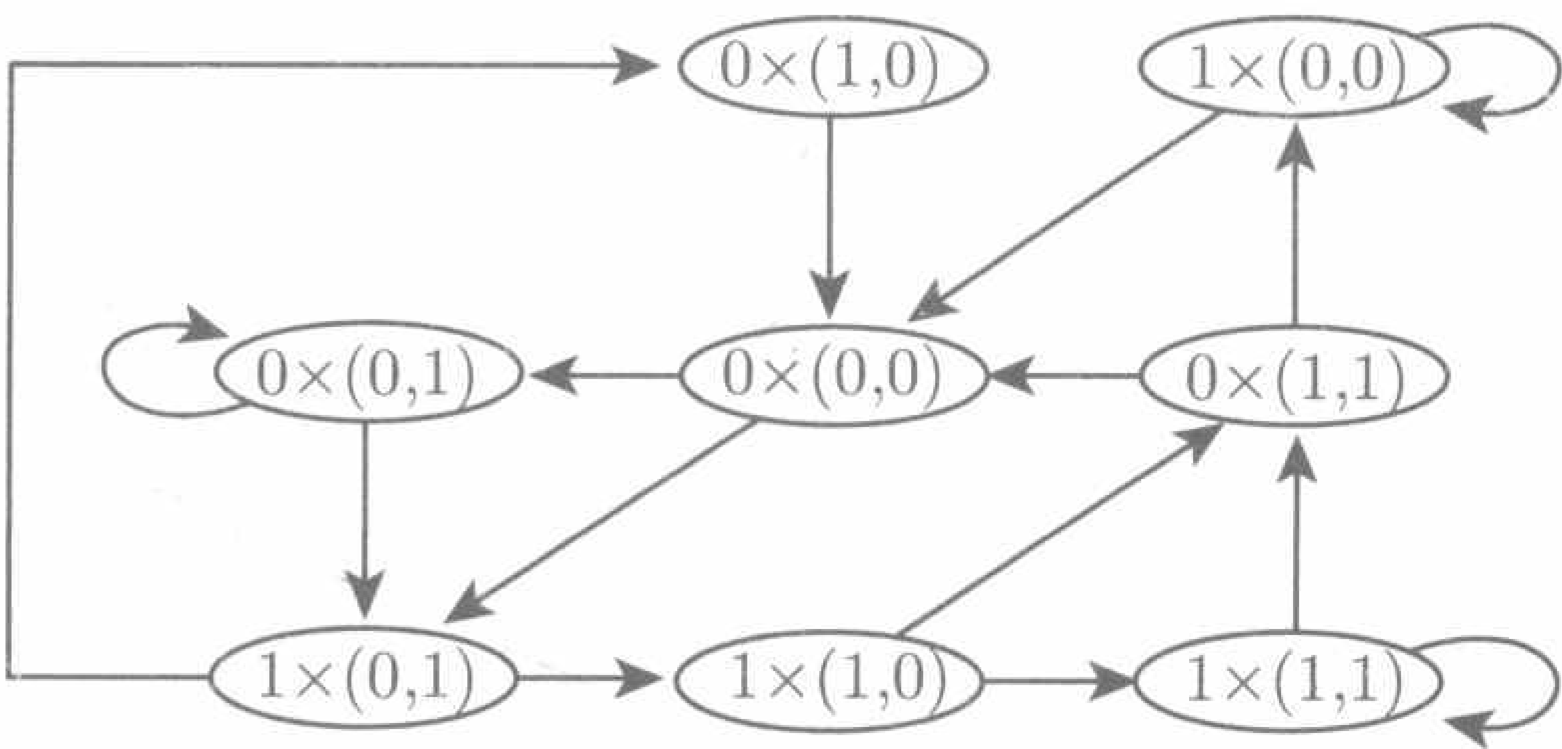


图 11.3.1　输入–状态转移图

现在, 输入–状态关联矩阵 $\mathcal{J} \in \mathcal{B}_{8\times 8}$ 可构造如下:

$$\mathcal{J}_{ij} = \begin{cases} 1, & [V_j, V_i] \in E, \\ 0, & \text{其他}. \end{cases}$$

不难得到

$$\mathcal{J} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}. \tag{11.3.4}$$

11.4　关联矩阵与能控能观性

实际上, 输入–状态关联矩阵是一个控制与状态的联合转移矩阵, 它显然可以用来刻画系统的动态过程, 譬如, 系统的能控性等.

在讨论其应用前, 我们先对输入–状态关联矩阵的结构进行一点分析, 从而发现 $\mathcal{J}$ 的一个简单算法. 比较 (11.3.3) 与 (11.3.4), 不难发现

$$\mathcal{J} = \begin{bmatrix} L \\ L \end{bmatrix}.$$

这是偶然的吗? 其实不是, 如果考虑到 $u(t+1)$ 可以自由选择, 即, 如果 $[u, x] \to [v, y]$, 则 $[u, x] \to [w, y]$, $\forall w \in \mathcal{D}^m$. 因此, 有

命题 11.4.1　设系统 (11.1.1) 的结构矩阵为 L(见 (11.1.2)), 那么, 它的输入–状

态关联矩阵为

$$\mathcal{J} = \left.\begin{bmatrix} L \\ L \\ \vdots \\ L \end{bmatrix}\right\} k^m, \tag{11.4.1}$$

这里 $\mathcal{J} \in \mathcal{B}_{k^{m+n} \times k^{m+n}}$.

定义 11.4.1　一个矩阵 $A \in \mathcal{M}_{m\times n}$ 称为以 τ 为周期的行周期矩阵, 如果 $\tau < m$ 为 m 的因子, 并且

$$\mathrm{Row}_{i+\tau}(A) = \mathrm{Row}_i(A), \quad 1 \leqslant i \leqslant m-\tau.$$

命题 11.4.2　1) 设 $A \in \mathcal{M}_{m\times m}$ 是周期为 r 的行周期矩阵 (记 $m = \tau r$), 当且仅当

$$A = \mathbf{1}_\tau \otimes A_0,$$

这里 $A_0 \in \mathcal{M}_{r\times m}$ 由 A 的首 r 行组成, 称为 A 的基块.

2) $A \in \mathcal{M}_{m\times m}$ 是周期为 r 的行周期矩阵, 则 A^s, $s \in \mathbb{Z}_+$, 也是周期为 r 的行周期矩阵.

记 $\mathcal{J}^s$ 的基块为 $\mathcal{J}_0^s$. 将 L 矩阵分为 k^m 块 $k^n \times k^n$ 的方阵如下:

$$L = [L_1\ L_2\ \cdots\ L_{2^m}],$$

然后定义

$$M := \sum_{i=1}^{k^m} L_i. \tag{11.4.2}$$

那么, 有如下公式计算 $\mathcal{J}^s$ 的基块:

命题 11.4.3

$$\mathcal{J}_0^{s+1} = M^s L. \tag{11.4.3}$$

实际上, 式 (11.4.2) 可直接用于检验系统的能控性.

定理 11.4.1　考虑系统 (11.1.1). 设 M 由 (11.4.2) 定义.

1) $x(s) = \delta_{k^n}^{\alpha}$ 是从 $x(0) = \delta_{2^n}^{j}$ 出发 s 步可达, 当且仅当

$$(M^s)_{\alpha j} > 0. \tag{11.4.4}$$

2) 定义能控性矩阵

$$\mathcal{C} := \sum_{s=1}^{k^{m+n}} M^s. \tag{11.4.5}$$

- $x = \delta_{k^n}^{\alpha}$ 是从 $x(0) = \delta_{k^n}^{j}$ 可达, 当且仅当

$$\mathcal{C}_{\alpha j} > 0. \tag{11.4.6}$$

- 系统在 $x(0) = \delta_{k^n}^{j}$ 能控, 当且仅当

$$\mathrm{Col}_j(\mathcal{C}) > 0. \tag{11.4.7}$$

- 系统能控, 当且仅当, 能控性矩阵为正矩阵, 即

$$\mathcal{C} > 0. \tag{11.4.8}$$

由前面的讨论可以看出, 在定义能控性矩阵时 $\mathcal{C}$ 阵元素的具体值无关结论, 为方便计, 可用 “$\vee$” 代替真正的加法. (即在布尔矩阵加法或乘法中令 $1+1=1$, 其他运算依旧.) 在这样意义下的矩阵幂 M^s 记作 $M^{(s)}$. 于是, 能控性矩阵可定义为

$$\mathcal{C} := \bigvee_{s=1}^{2^{m+n}} M^{(s)}. \tag{11.4.9}$$

则定理 11.4.1 的结论依旧成立. 下面是一个简单例子:

例 11.4.1 考虑一个布尔控制网络

$$\begin{cases} x_1(t+1) = (x_1(t) \leftrightarrow x_2(t)) \vee u_1(t), \\ x_2(t+1) = \neg x_1(t) \wedge u_2(t). \end{cases} \tag{11.4.10}$$

记 $x(t) = \ltimes_{i=1}^2 x_i(t)$, $u = \ltimes_{i=1}^2 u_i(t)$, 有

$$x(t+1) = Lu(t)x(t), \tag{11.4.11}$$

这里

$$L = \delta_4[2\ 2\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 4\ 3\ 1\ 2\ 4\ 4\ 2].$$

于是

$$M = \bigvee_{i=1}^{4} L_i = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}.$$

1) 从 $x(0)=\delta_4^2$ 出发, 能否到达 δ_4^1? 直接计算可得

$$(M^{(1)})_{12}=0,\quad (M^{(2)})_{12}>0.$$

因此, $x(2)=\delta_4^1$ 由 $x(0)=\delta_4^2$ 出发 2 步可达.

2) 系统在哪些点可控, 系统是否可控?

系统的能控性矩阵为

$$\mathcal{C}=\sum_{s=1}^{2^4}M^{(s)}=\begin{bmatrix}1&1&1&1\\1&1&1&1\\0&0&1&0\\1&1&1&1\end{bmatrix}.$$

根据定理 11.4.1, 有

- 系统不是完全可控的, 但它在 $x_0=\delta_4^3\sim(0,1)$ 可控.
- 如果从 $x_0=\delta_4^1\sim(1,1)$, 或 $x_0=\delta_4^2\sim(1,0)$, 或 $x_0=\delta_4^4\sim(0,0)$ 出发, 都不能到达 $x_d=\delta_4^3\sim(0,1)$.

下面介绍用输入–状态关联矩阵讨论能观性的结果. 先做一点记号上的准备.

定义 11.4.2　对布尔矩阵 $A=(a_{ij})\in\mathcal{B}_{m\times n}$, 其布尔权重为

$$w_b(A):=\begin{cases}0, & A=0,\\1, & \text{其他}.\end{cases}\tag{11.4.12}$$

定义

$$\tilde{\mathcal{J}}_0^{(s)}:=\mathcal{J}_0^{(s)}W_{[k^n,k^m]},\tag{11.4.13}$$

将它分为 k^n 个 $k^n\times k^m$ 的块矩阵如下:

$$\tilde{\mathcal{J}}_0^{(s)}=\left[\left(\tilde{\mathcal{J}}_0^{(s)}\right)_1\ \left(\tilde{\mathcal{J}}_0^{(s)}\right)_2\ \cdots\ \left(\tilde{\mathcal{J}}_0^{(s)}\right)_{k^n}\right],\tag{11.4.14}$$

这里 $\left(\tilde{\mathcal{J}}_0^{(s)}\right)_i\in\mathcal{B}_{k^n\times k^m}$, $i=1,\cdots,k^n$.

下面给出能观的主要结果:

定理 11.4.2　考虑系统 (11.1.1), 其代数状态空间表达式为 (11.1.2). 如果

$$\bigvee_{s=1}^{2^{m+n}}\left[\left(H\ltimes\left(\tilde{\mathcal{J}}_0^{(s)}\right)_i\right)\bar{\vee}\left(H\ltimes\left(\tilde{\mathcal{J}}_0^{(s)}\right)_j\right)\right]\neq 0,\quad 1\leqslant i<j\leqslant k^n,\tag{11.4.15}$$

则系统能观.

以下的推论是一个等价的表达形式.

推论 11.4.1 考虑系统 (11.1.1), 其代数状态空间表达式为式 (11.1.2). 记

$$O_{ij} := \bigvee_{s=1}^{2^{m+n}} \left[\left(H \ltimes \left(\tilde{\mathcal{J}}_0^{(s)} \right)_i \right) \bar{\vee} \left(H \ltimes \left(\tilde{\mathcal{J}}_0^{(s)} \right)_j \right) \right].$$

如果

$$\wedge_{1 \leqslant i < j \leqslant 2^n} w_b(O_{ij}) = 1, \tag{11.4.16}$$

则系统可观.

与定理 11.2.1 相比, 定理 11.4.2 (或推论 11.4.1) 只是充分条件, 但后者不要求系统能控.

下面给一个例子.

例 11.4.2 考虑例 11.4.1 中的系统 (11.4.10). 设输出为

$$y(t) = x_1(t) \vee x_2(t). \tag{11.4.17}$$

于是

$$y(t) = Hx(t) = \delta_2[1\ 1\ 1\ 2]x(t). \tag{11.4.18}$$

记

$$O_{ij} = \bigvee_{s=1}^{2^4} \left[\left(H \ltimes \left(\tilde{\mathcal{J}}_0^{(s)} \right)_i \right) \bar{\vee} \left(H \ltimes \left(\tilde{\mathcal{J}}_0^{(s)} \right)_j \right) \right].$$

简单计算可得

$$O_{12} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad O_{13} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \quad O_{14} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix},$$

$$O_{23} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \quad O_{24} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}, \quad O_{34} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

因此有

$$\bigwedge_{1 \leqslant i < j \leqslant 4} w_b(O_{ij}) = 1.$$

故系统能观.

11.5　注释与参考

能控性与能观测性是控制系统最基本的两个要素, k 值逻辑控制网络也不例外. 目前, 对布尔控制网络的能控性与能观测性的研究最为深入, 成果丰富. 本章仅介绍基本概念与结论. 本文的结果主要来自文献 [52, 153]. 这方面的后续研究很多, 例如文献 [74, 94, 102, 103] 等.

将 k 值逻辑控制网络能控性与能观测性的结果推广到多值的情况在原则上应当是不难的, 但目前尚未见到具体结果.

第 12 章　逻辑系统的干扰解耦

12.1　干扰解耦的动态模型

带干扰的 k 值逻辑控制系统的动态方程可表示如下:

$$
\begin{cases}
x_1(t+1) = f_1(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t),\xi_1(t),\cdots,\xi_q(t)),\\
\qquad\vdots\\
x_n(t+1) = f_n(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t),\xi_1(t),\cdots,\xi_q(t)),
\end{cases}
$$
$$
y_j(t) = h_j(x(t)),\quad j=1,\cdots,p, \tag{12.1.1}
$$

这里 $\xi_i(t)$, $i=1,\cdots,q$ 为干扰. 记 $x(t)=\ltimes_{i=1}^n x_i(t)$, $u(t)=\ltimes_{i=1}^m u_i(t)$, $\xi(t)=\ltimes_{i=1}^q \xi_i(t)$, $y(t)=\ltimes_{i=1}^p y_i(t)$. 那么, (12.1.1) 的代数状态空间表示为

$$
\begin{cases}
x(t+1) = Lu(t)\xi(t)x(t),\\
y(t) = Hx(t),
\end{cases} \tag{12.1.2}
$$

这里 $L\in\mathcal{L}_{k^n\times k^{n+m+q}}$, $H\in\mathcal{L}_{k^p\times k^n}$.

所谓干扰解耦, 就是要找一个合适的控制, 使得干扰不影响系统的输出. 先用一个简单例子来刻画一下.

例 12.1.1　考察如下的带干扰的布尔控制网络

$$
\begin{cases}
x_1(t+1) = x_2(t)\wedge\xi(t),\\
x_2(t+1) = x_3(t)\vee u_1(t),\\
x_3(t+1) = x_4(t)\wedge[(x_2(t)\to\xi(t))\vee u_1(t)],\\
x_4(t+1) = \neg x_3(t)\vee[\xi(t)\wedge u_2(t)],
\end{cases}
$$
$$
y(t) = x_3(t)\wedge x_4(t). \tag{12.1.3}
$$

如果选择控制

$$
u_1(t) = x_2(t),\quad u_2(t) = 0,
$$

于是, 闭环系统变为

$$
\begin{cases}
x_1(t+1) = x_2(t) \wedge \xi(t),\\
x_2(t+1) = x_3(t) \vee x_2(t),\\
x_3(t+1) = x_4(t),\\
x_4(t+1) = \neg x_3(t),
\end{cases}
\quad (12.1.4)
$$
$$
y(t) = x_3(t) \wedge x_4(t).
$$

显然, 干扰不再影响输出.

我们给出如下严格定义:

定义 12.1.1 考察系统 (12.1.1). 干扰解耦问题指: 寻找状态反馈控制

$$
u(t) = \phi(x(t)), \quad (12.1.5)
$$

以及一个坐标变换 $z = T(x)$, 使在 z 坐标下闭环系统变为

$$
\begin{cases}
z^1(t+1) = F^1(z(t), \phi(x(t)), \xi(t)),\\
z^2(t+1) = F^2(z^2(t)),
\end{cases}
\quad (12.1.6)
$$
$$
y(t) = G(z^2(t)).
$$

由定义 12.1.1 可知, 要解干扰解耦问题需要解决两个问题: (i) 找到一个正规子空间 z^2, 它涵盖了输出; (ii) 设计一个控制和 z^2 的补空间 z^1, 使干扰在 z^2 的动力学方程中消失. 我们依次解决这两个问题.

12.2 Y 友好子空间

定义 12.2.1 设 $\mathcal{X} = \mathcal{F}_\ell\{x_1, \cdots, x_n\}$ 是系统的状态空间, $Y = \{y_1, \cdots, y_p\} \subset \mathcal{X}$. 一个正规子空间 $\mathcal{Z} \subset \mathcal{X}$ 称为 Y 友好子空间, 如果 $y_i \in \mathcal{Z}$, $i = 1, \cdots, p$. 一个具有最小维数的 Y 友好子空间称为最小 Y 友好子空间.

如果 Y 是系统输出, Y 友好子空间也称输出友好子空间. 下面讨论如何构造 (最小)Y 友好子空间.

设 $Y = \{y_1, \cdots, y_p\} \subset \mathcal{X}$ 为 p 个 k 值逻辑函数, 定义 $y = \ltimes_{i=1}^p y_i$. 那么, y 在代数形式下为

$$
y = \delta_{k^p}[i_1, i_2, \cdots, i_{k^n}]x := Hx. \quad (12.2.1)
$$

记 H 中 j 的个数为 n_j, 即

$$
n_j = \big|\{s \,|\, i_s = j, 1 \leqslant s \leqslant k^n\}\big|, \quad j = 1, \cdots, k^p.
$$

其中 $|\cdot|$ 表示表示集合的势 (元素个数). 那么, 有如下结论:

定理 12.2.1 设 $y = \ltimes_{i=1}^{p} y_i$, 其代数表达式为 (12.2.1).

1) 存在 r 维 Y 友好子空间, 当且仅当, $n_j, j = 1, \cdots, k^p$ 具有公共的 (k 型) 因子 k^{n-r}.

2) 设 k^{n-r} 为 $n_j, j = 1, \cdots, k^p$, 具有的最大的公共 (k 型) 因子, 那么, 最小 Y 友好子空间维数为 r.

下面, 给出一个算法来构造 Y 友好子空间: 设 k^{n-r} 为 n_i 的公共因子, 记 $n_i = m_i \cdot k^{n-r}$, $i = 1, \cdots, k^p$. 将 H 的列 $\mathrm{Col}(H)$ 分为 k^p 个子集 J_i, $i = 1, \cdots, k^p$. $\rho \in J_i$ 表明 H 的第 ρ 列是 $\mathrm{Col}_\rho(H) = \delta_{k^p}^i$.

利用相应的结构矩阵不难看出, 寻找 Y 友好子空间就等于构造一个正规子空间的逻辑矩阵 $T_0 \in \mathcal{L}_{k^r \times k^n}$, 使我们能够找到一个 $G \in \mathcal{L}_{k^p \times k^r}$, 使得

$$GT_0 = H. \tag{12.2.2}$$

下面给出算法:

算法 12.2.1 • 步骤 1. 将 T_0 的行按如下方法分成 k^p 块: I_1 由首 m_1 行组成, I_2 为接下来的 m_2 行组成, 直至 I_{k^p} 由最后的 m_{k^p} 行组成. (注意, $\sum_{i=1}^{k^p} m_i = k^r$.) 将 T_0 元素分成 $k^p \times k^p$ 子集如下

$$T_0^{i,j} = \{t_{r,s} \,|\, r \in I_i, s \in J_j\}, \quad i, j = 1, \cdots, k^p.$$

(注意, $T_0^{i,j}$ 一般不是子块, 因 J_j 不连续.)

• 步骤 2. 将 $T_0^{i,j}$ 看成一个 $m_i \times (m_j k^{n-r})$ 子矩阵. 定义其值为

$$T_0^{i,j} = \begin{cases} I_{m_i} \otimes \mathbf{1}_{k^{n-r}}^{\mathrm{T}}, & i = j, \\ 0, & \text{其他}. \end{cases} \tag{12.2.3}$$

其中 I_{m_i} 为 m_i 阶单位矩阵.

• 步骤 3. 设

$$z = \ltimes_{i=1}^{r} z_i := T_0 x.$$

反算出 $z_i, i = 1, \cdots, r$.

实际上, 算法 12.2.1 可视为对定理 12.2.1 的构造性证明.

命题 12.2.1 设 k^{n-r} 为 $n_i, i = 1, \cdots, k^p$, 的公共因子. 那么, 由算法 12.2.1 产生的 $z_i, i = 1, \cdots, r$ 形成 r 维 Y 友好子空间.

证明　构造

$$G = \begin{bmatrix} \mathbf{1}_{m_1}^{\mathrm{T}} & 0 & \cdots & 0 \\ 0 & \mathbf{1}_{m_2}^{\mathrm{T}} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \mathbf{1}_{m_{k^p}}^{\mathrm{T}} \end{bmatrix}. \tag{12.2.4}$$

根据 T_0 的构造, 不难验证

$$y = GT_0x = Gz. \tag{12.2.5}$$

□

我们特别感兴趣的是, 如何构造最小的 Y 友好子空间, 下面用一个例子说明这个构造方法.

例 12.2.1　设 $\mathcal{X} = \mathcal{F}_\ell\{x_1, x_2, x_3\}$, 考虑 $k=3$ 的情况.

$$\begin{aligned} y_1 &= f_1(x_1,x_2,x_3) = (x_1 \leftrightarrow x_3) \wedge x_2, \\ y_2 &= f_2(x_1,x_2,x_3) = x_1 \wedge x_3. \end{aligned} \tag{12.2.6}$$

注意, 此时变量取值于 $\mathcal{D}_3 = \{0,\ 0.5,\ 1\}$, 逻辑运算

$$x \leftrightarrow y = \begin{cases} 1, & x = y, \\ 0, & \text{其他}, \end{cases}$$

$$x \wedge y = \min\{x, y\}.$$

令 $0 \sim \delta_3^3$, $0.5 \sim \delta_3^2$, 及 $1 \sim \delta_3^1$.

我们寻找最小 Y 友好子空间. 记 $y = y_1y_2$, $x = x_1x_2x_3$, 容易算得

$$\begin{aligned} y_1 &= M_cM_ex_1x_3x_2 \\ &= M_cM_e(I_3 \otimes W_{[3]})x_1x_3x_2 \\ &:= M_1x, \end{aligned}$$

于是, f_1 的结构矩阵 M_1 为

$$\begin{aligned} M_1 &= M_cM_e(I_3 \otimes W_{[3]}) \\ &= \delta_3[1\ 2\ 3\ 2\ 2\ 3\ 3\ 3\ 3\ 2\ 2\ 2\ 2\ 2\ 2\ 3\ 3\ 3\ 3\ 2\ 1\ 3\ 2\ 2\ 3\ 3\ 3]. \end{aligned}$$

类似地, 可算得 $y_2 = M_2x$, 这里

$$M_2 = \delta_3[1\ 2\ 3\ 1\ 2\ 3\ 1\ 2\ 3\ 2\ 2\ 3\ 2\ 2\ 3\ 2\ 2\ 3\ 3\ 3\ 3\ 3\ 3\ 3\ 3\ 3\ 3].$$

最终可得

$$y = Mx,$$

这里

$$M = \delta_9[1\ 5\ 9\ 4\ 5\ 9\ 7\ 8\ 9\ 5\ 5\ 6\ 5\ 5\ 6\ 8\ 8\ 9\ 9\ 6\ 3\ 9\ 6\ 6\ 9\ 9\ 9].$$

由 M 可算出 $n_1 = n_3 = n_4 = n_7 = 1, n_2 = 0, n_5 = 6, n_6 = 5, n_8 = 3$ 及 $n_9 = 9$. 它们的公共 3 型因子为 3^{n-r}, 于是, 最小 Y 友好子空间的维数为 $r = 3$.

为构造 T_0, 有

$$\begin{aligned}
&J_1 = \{1\}; \quad J_2 = \varnothing; \quad J_3 = \{21\}; \quad J_4 = \{4\};\\
&J_5 = \{2, 5, 10, 11, 13, 14\}; \quad J_6 = \{12, 15, 20, 23, 24\};\\
&J_7 = \{7\}; \quad J_8 = \{8, 16, 17\}; \quad J_9 = \{3, 6, 9, 18, 19, 22, 25, 26, 27\}.
\end{aligned}$$

因为 $m_i = n_i, i = 1, \ldots, 9$, 则 $I_1 = \{1\}$, $I_2 = \varnothing$, $I_3 = \{2\}$, $I_4 = \{3\}$, $I_5 = \{4, 5, 6, 7, 8, 9\}$, $I_6 = \{10, 11, 12, 13, 14\}$, $I_7 = \{15\}$, $I_8 = \{16, 17, 18\}$, $I_9 = \{19, 20, 21, 22, 23, 24, 25, 26, 27\}$. 设 $T_0^{1,1}$ 为 $\mathbf{1}^{\mathrm{T}}$, 不难知道 T_0 的第 1 为 δ_{27}^1. 类此, T_0 的第 21 列为 δ_{27}^9, 第 4 列为 δ_{27}^2 等. 最后, T_0 可算得如下:

$$\begin{aligned}
T_0 = \delta_{27}[&1\ 21\ 4\ 2\ 5\ 10\ 11\ 13\ 14\ 12\ 15\ 20\ 23\ 24\ 7\\
&8\ 16\ 17\ 3\ 6\ 9\ 18\ 19\ 22\ 25\ 26\ 27].
\end{aligned} \tag{12.2.7}$$

G 可由公式 (12.2.4) 算出:

$$G = \delta_9[1\ 2\ 3\ 4\ 4\ 4\ 4\ 4\ 4\ 5\ 5\ 5\ 5\ 5\ 6\ 7\ 7\ 7\ 8\ 8\ 8\ 8\ 8\ 8\ 8\ 8\ 8]. \tag{12.2.8}$$

最后, 设最小 Y 友好子空间由 $\{z_1, z_2, z_3\}$ 生成. 记 $z = z_1z_2z_3$, 则得

$$z = T_0x.$$

记 $z_i := E_ix$, $i = 1, 2, 3$. 不难从 T_0 算出 z_i 的结构矩阵 E_i 如下:

$$\begin{aligned}
E_1 &= \delta_3[1\ 3\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3\ 3\ 3\ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 2\ 3\ 3\ 3\ 3\ 3];\\
E_2 &= \delta_3[1\ 1\ 2\ 1\ 2\ 1\ 1\ 2\ 2\ 1\ 2\ 1\ 2\ 2\ 3\ 3\ 3\ 3\ 1\ 2\ 3\ 3\ 1\ 2\ 3\ 3\ 3];\\
E_3 &= \delta_3[1\ 3\ 1\ 2\ 2\ 1\ 2\ 1\ 2\ 3\ 3\ 2\ 2\ 3\ 1\ 2\ 1\ 2\ 3\ 3\ 3\ 3\ 1\ 1\ 1\ 2\ 3].
\end{aligned} \tag{12.2.9}$$

同样, 不难算出

$$\begin{aligned}
y_1 &= \delta_3[1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3\ 3]z;\\
y_2 &= \delta_3[1\ 2\ 3\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 3\ 1\ 2]z.
\end{aligned}$$

12.3 解耦控制设计

12.2 节讨论了 Y 友好子空间的计算. 现在假定一个 Y 友好子空间 $\mathcal{Z}=\mathcal{F}_\ell\{z^2\}$ 已给定. 那么, 就可以找到 z^1, 使 $z=\{z^1,z^2\}$ 为一新坐标系, 并且, 在 z 坐标下, 系统 (12.1.1) 可表示为

$$\begin{cases}z^1(t+1)=F^1(z(t),u(t),\xi(t)),\\ z^2(t+1)=F^2(z(t),u(t),\xi(t)),\end{cases}\\ y(t)=G(z^2(t)). \tag{12.3.1}$$

式 (12.3.1) 称为输出友好型.

比较 (12.3.1) 与 (12.1.6), 不难看出, 要解决干扰解耦问题, 就是要寻找反馈控制 $u(t)=u(z(t))$ 使得

$$F^2(z(t),u(z(t)),\xi(t))=\tilde{F}^2(z^2(t)). \tag{12.3.2}$$

下面考虑控制设计. 首先考虑布尔函数, 即 $k=2$. 对于 $k>2$ 的情况不难从 $k=2$ 的结论平行推出.

先给一个命题, 它可用于将布尔系统从代数状态空间表示返回到逻辑表示.

命题 12.3.1 设一个逻辑变量 E, 其代数表达式为

$$E=f(x_1,x_2,\cdots,x_n)=M_fx_1x_2\cdots x_n, \tag{12.3.3}$$

这里 $M_f\in\mathcal{L}_{2\times 2^n}$ 为 f 的结构矩阵. 那么,

$$E=[x_1\wedge f_1(x_2,\cdots,x_n)]\vee[\neg x_1\wedge f_2(x_2,\cdots,x_n)], \tag{12.3.4}$$

这里

$$M_f=(M_{f_1}\ |M_{f_2}),$$

即, f_1 及 f_2 的结构矩阵分别为 M_f 的前、后半部.

证明 利用 (12.3.3), 当 $x_1=1$ 时,

$$E=M_f\delta_2^1x_2\cdots x_n=M_{f_1}x_2\cdots x_n.$$

而当 $x_1=0$ 时,

$$E=M_f\delta_2^2x_2\cdots x_n=M_{f_2}x_2\cdots x_n.$$

因此, 可得 (12.3.4). □

设 $z^2=(z^2_1,\cdots,z^2_\ell)$, $\mathcal{Z}^2:=\mathcal{F}_\ell\{z^2\}$ 为 ℓ 维子空间, 定义

$$
\begin{aligned}
&e_1(z^2)=z^2_1\wedge z^2_2\wedge\cdots\wedge z^2_\ell;\qquad e_2(z^2)=z^2_1\wedge z^2_2\wedge\cdots\wedge\neg z^2_\ell;\\
&e_3(z^2)=z^2_1\wedge\cdots\wedge\neg z^2_{\ell-1}\wedge z^2_\ell; e_4(z^2)=z^2_1\wedge\cdots\wedge\neg z^2_{\ell-1}\wedge\neg z^2_\ell;\\
&\qquad\cdots\qquad\qquad\qquad e_{2^\ell}(z^2)=\neg z^2_1\wedge\neg z^2_2\wedge\cdots\wedge\neg z^2_\ell.
\end{aligned}\tag{12.3.5}
$$

则

$$
\mathcal{Z}^2=\mathcal{F}_\ell\{e_i\,|\,1\leqslant i\leqslant 2^\ell\}.
$$

称 $\{e_i\,|\,1\leqslant i\leqslant 2^\ell\}$ 为 $\mathcal{Z}^2$ 的合取基.

利用命题 12.3.1 不难看出, F^2 中的每一个方程, 记作 F^2_j, $j=1,\cdots,\ell$, 可表示成

$$
F^2_j(z(t),u(t),\xi(t))=\vee_{i=1}^{2^\ell}[e_i(z^2(t))\wedge Q^i_j(z^1(t),u(t),\xi(t))],\quad j=1,\cdots,\ell.\tag{12.3.6}
$$

利用表达形式 (12.3.6) 可得下面的命题:

命题 12.3.2 考虑 $k=2$. $F^2(z(t),u(t),\xi(t))=F^2(z^2(t))$, 当且仅当, 在表达式 (12.3.6) 中,

$$
Q^i_j(z^1(t),u(t),\xi(t))=\text{const.},\quad j=1,\cdots,\ell;\ i=1,\cdots,2^p.\tag{12.3.7}
$$

证明 充分性的证明是平凡的. 对于必要性, 假设对于一个特殊对 i,j, Q^i_j 不是常数. 考虑相应的 e_i, 如果其 z^2_w 因子是 z^2_w, 则令 $z^2_w=1$. 否则, 如果其因子为 $\neg z^2_w$, 令 $z^2_w=0, w=1,2,\ldots,\ell$. 则有

$$
e_i(z^2)=1,\quad e_j(z^2)=0,\quad j\neq i.
$$

由于 Q^i_j 不是常数, 当 $Q^i_j=1$ 时, 有 $F^2_j=1$; 当 $Q^i_j=0$ 时, 有 $F^2_j=0$. 因此, 对于固定的 z^2, F^2_j 有不同的值, 这意味着 F^2_j 不只是 z^2 的函数. □

下面讨论 $k>2$ 的情况. 先将表达式 (12.3.6) 推广到 $k>2$ 的情况. 设 $x\in\mathcal{D}_k=\{0,1,\cdots,k-1\}$, 其向量形式为: $i\sim\delta_k^{i+1}$, $i=0,1,\cdots,k-1$. 定义一个布尔值函数

$$
\pi(x):=\begin{cases}1, & x=0\\ 0, & x\neq 0.\end{cases}
$$

定义 $a+_k b:=(a+b)(\mathrm{mod}\ k)$.

考虑一个 k 值逻辑函数

$$
f(x_1,\cdots,x_n)=Mx,\tag{12.3.8}
$$

这里 $x = \ltimes_{i=1}^{n} x_i$. 将 M 均分为 k 块, 记作

$$M = [M_1, M_2, \cdots, M_k],$$

这里 $M_i \in \mathcal{L}_{k \times k^{n-1}}$, $i = 1, \cdots, k$. 那么, 相应于命题 12.3.1, 有

命题 12.3.3

$$f(x_1, \cdots, x_n) = \sum_{i=1}^{k} \pi(x_1 +_k (k-i)) \wedge f_i(x_2, \cdots, x_{n-1}), \tag{12.3.9}$$

这里, f_i 的结构矩阵为 M_i, $i = 1, \cdots, k$.

反复使用命题 12.3.3, 则可得到与 (12.3.6) 相应的表达式

$$\begin{aligned} & F_j^2(z(t), u(t), \xi(t)) \\ = & \wedge_{\alpha_1=0}^{k-1} \cdots \wedge_{\alpha_\ell=0}^{k-1} \pi(z_1^2 +_k (k-\alpha_1)) \cdots \pi(z_\ell^2 +_k (k-\alpha_\ell)) \\ & \wedge Q_j^{\alpha_1, \cdots, \alpha_\ell}(z^1(t), u(t), \xi(t)), \quad j = 1, \cdots, \ell. \end{aligned} \tag{12.3.10}$$

不难将命题 12.3.2 推广到 $k > 2$ 的情况:

命题 12.3.4 *考虑 $k > 2$. $F^2(z(t), u(t), \xi(t)) = F^2(z^2(t))$, 当且仅当, 在表达式 (12.3.10) 中,*

$$Q_j^{\alpha_1, \cdots, \alpha_\ell}(z^1(t), u(t), \xi(t)) = \text{const.}, \quad \alpha_j = 0, \cdots, k-1,\ j = 1, \cdots, \ell. \tag{12.3.11}$$

综合以上讨论可得到如下结果:

定理 12.3.1 *考察系统 (12.1.1). 干扰解耦问题可解, 当且仅当,*

(i) 存在一个 Y 友好子空间, 使系统在这个 Y 友好坐标下表示为形如 (12.3.1) 的输出友好型;

(ii) 在 (12.3.1) 中, 将 F^2 表示为标准形式: (准确地说, 当 $k = 2$ 时为 (12.3.6); $k > 2$ 时为 (12.3.10)), 存在控制 $u(t) = u(z(t))$ 使 Q_j 为定常 (准确地说, 当 $k = 2$ 时 (12.3.7) 成立, $k > 2$ 时 (12.3.11) 成立).

注 *获得了输出友好坐标的子基, 即 z^2 之后, 必须选择逻辑变量的补集 z^1 以形成一个新的坐标系 $z = \{z^1, z^2\}$. 很容易检验, z^1 的选择并不影响干扰解耦问题的可解性.*

下面假设已经得到式 (12.3.1) 以及 z^2 的最小输出友好子空间, 则可以寻找反馈控制. 设反馈控制为

$$u(z^1(t)) = U z^1(t). \tag{12.3.12}$$

其中 $U \in \mathcal{L}_{k^m \times k^{n-\ell}}$.

注意, 只有有限多的矩阵 U. 如果存在一个控制 u 使得式 (12.3.7) 中的所有函数为常数, 则我们实现了干扰解耦的目标. 否则, 不存在 u 可以从式 (12.3.7) 中的所有函数中消去 ξ, 这意味着干扰解耦问题是不可解的. 如果存在一个 u 虽然不能从式 (12.3.7) 中的所有函数中消去 ξ, 但是有一部分 z^1 的函数, 即 $\{\eta_1,\ldots,\eta_t\}\subset\mathcal{F}_\ell\{z^1\}$, 其中仍有 ξ 未被消除, 则增加 $\{\eta_i|i=1,\ldots,t\}$ 到 $\{y_1,\ldots,y_p\}$ 中, 并且找到一个最小 $(y_i,\eta_j|i=1,\ldots,p;j=1,\ldots,t)$ 友好子空间, 记为 V. 则我们将闭环系统转化为式 (12.3.1) 的形式, 其中 $\mathcal{F}_\ell\{z^2\}=V$, 然后检查它是否具有式 (12.1.6) 的形式.

检验一个带有输出友好子空间的可能解能否获得, 首先寻找一个最小子空间, 我们必须说明一个输出友好子空间包含一个最小输出友好子空间. 我们有下面的结果.

命题 12.3.5 令 V 为一个 $Y=\{y_1,\cdots,y_p\}$ 友好子空间. 则存在一个最小 Y 友好子空间, $W\subset V$.

证明 设

$$y=\ltimes_{i=1}^p y_i=Hx,$$

令 n_i, $i=1,\cdots,k^p$ 表示 H 的列数, H 的列为 $\delta_{k^p}^i$. 令 k^s 为 $\{n_i\}$ 的最大公共 k-型因子. 则最小 Y 友好子空间的维数为 $n-s$. 令 $\{v_1,\cdots,v_t\}$ 为 V 的一个基. 如果 $t=n-s$, 则命题得证. 因此, 假设 $t>n-s$. 由于 V 是一个 Y 友好子空间, 设 $v=\ltimes_{i=1}^t v_i$, 我们可以表示

$$y=Gv,\quad 其中\ G\in\mathcal{L}_{k^p\times k^t}. \tag{12.3.13}$$

设 r_i, $i=1,\cdots,k^p$ 表示 G 的列数, 其等于 $\delta_{k^p}^i$. 令 k^j 为 r_i 的最大公共 k-型因子, $r_i=m_ik^j$. 由于 V 是一个正规子空间, 可表示为 $v=Ux$, 其中 $U\in\mathcal{L}_{k^t\times k^n}$. 由于 v 是一个正规子基, 则定理 10.3.1 的条件成立. 注意

$$y=Gv=GUx,$$

我们必须计算 GU. 利用 G 的构造以及 U 性质 (即定理 10.3.1 的条件), 很容易验证 $\delta_{k^p}^i$ 的每一列得出 GU 中 $\delta_{k^p}^i$ 的 k^{n-t} 列. 因此, 有 GU 中的 $\delta_{k^p}^i$,

$$r_i\cdot k^{n-t}=m_i\cdot k^{n-t+j},$$

其中 $i=1,\cdots,2^p$. 这意味着, $\{n_i\}$ 的最大公共 k 型因子为 k^{n-t+j}. 从而 $n-t+j=s$. 等价地,

$$j=t-(n-s).$$

回到式 (12.3.13), 由于 V 的维数为 t 并且 r_i 具有最大公共 k 型因子 k^j, 可以找到 V 的一个最小 Y 友好子空间, 维数为 $t-j=t-[t-(n-s)]=n-s$. 从维数可知, 该最小 Y 友好子空间也是 $\mathcal{X}=\mathcal{F}_\ell\{x_1,\cdots,x_n\}$ 的最小 Y 友好子空间. 命题得证. □

我们用一个例子说明解耦过程.

例 12.3.1 考虑 $k=2$ 的情况, 考察下列系统

$$
\begin{cases}
x_1(t+1)=x_4(t)\bar{\vee}u_1(t),\\
x_2(t+1)=(x_2(t)\bar{\vee}x_3(t))\wedge\neg\xi(t),\\
x_3(t+1)=[(x_2(t)\leftrightarrow x_3(t))\vee\xi(t)]\bar{\vee}[(x_1\leftrightarrow x_5)\vee u_2(t)],\\
x_4(t+1)=[u_1(t)\to(\neg x_2(t)\vee\xi(t))]\wedge(x_2(t)\leftrightarrow x_3(t)),\\
x_5(t+1)=(x_4(t)\bar{\vee}u_1(t))\leftrightarrow[(u_2(t)\wedge\neg x_2(t))\vee x_4(t)],
\end{cases}
\\
y(t)=x_4(t)\wedge(x_1(t)\leftrightarrow x_5(t)), \tag{12.3.14}
$$

这里 $u_1(t),u_2(t)$ 为控制, $\xi(t)$ 为干扰, $y(t)$ 为输出.

记 $x(t)=\ltimes_{i=1}^5 x_i(t)$, $u=u_1(t)u_2(t)$, 将 (12.3.14) 转化为代数形式

$$
\begin{aligned}
&x(t+1)=Lu(t)\xi(t)x(t),\\
&y(t)=Hx(t),
\end{aligned} \tag{12.3.15}
$$

这里

$$
\begin{aligned}
L=\delta_{32}[&30\ 30\ 14\ 14\ 32\ 32\ 16\ 16\ 32\ 32\ 15\ 15\ 30\ 30\ 13\ 13\\
&30\ 30\ 14\ 14\ 32\ 32\ 16\ 16\ 32\ 32\ 15\ 15\ 30\ 30\ 13\ 13\\
&32\ 32\ 16\ 16\ 20\ 20\ 4\ 4\ 20\ 20\ 3\ 3\ 30\ 30\ 13\ 13\\
&32\ 32\ 16\ 16\ 20\ 20\ 4\ 4\ 20\ 20\ 3\ 3\ 30\ 30\ 13\ 13\\
&30\ 26\ 14\ 10\ 32\ 28\ 16\ 12\ 32\ 28\ 16\ 12\ 30\ 26\ 14\ 10\\
&26\ 30\ 10\ 14\ 28\ 32\ 12\ 16\ 28\ 32\ 12\ 16\ 26\ 30\ 10\ 14\\
&32\ 28\ 16\ 12\ 20\ 24\ 4\ 8\ 20\ 24\ 4\ 8\ 30\ 26\ 14\ 10\\
&28\ 32\ 12\ 16\ 24\ 20\ 8\ 4\ 24\ 20\ 8\ 4\ 26\ 30\ 10\ 14\\
&13\ 13\ 29\ 29\ 15\ 15\ 31\ 31\ 15\ 15\ 32\ 32\ 13\ 13\ 30\ 30\\
&13\ 13\ 29\ 29\ 15\ 15\ 31\ 31\ 15\ 15\ 32\ 32\ 13\ 13\ 30\ 30\\
&13\ 13\ 29\ 29\ 3\ 3\ 19\ 19\ 3\ 3\ 20\ 20\ 13\ 13\ 30\ 30\\
&13\ 13\ 29\ 29\ 3\ 3\ 19\ 19\ 3\ 3\ 20\ 20\ 13\ 13\ 30\ 30\\
&13\ 9\ 29\ 25\ 15\ 11\ 31\ 27\ 15\ 11\ 31\ 27\ 13\ 9\ 29\ 25\\
&9\ 13\ 25\ 29\ 11\ 15\ 27\ 31\ 11\ 15\ 27\ 31\ 9\ 13\ 25\ 29\\
&13\ 9\ 29\ 25\ 3\ 7\ 19\ 23\ 3\ 7\ 19\ 23\ 13\ 9\ 29\ 25
\end{aligned}
$$

$$9 \quad 13 \quad 25 \quad 29 \quad 7 \quad 3 \quad 23 \quad 19 \quad 7 \quad 3 \quad 23 \quad 19 \quad 9 \quad 13 \quad 25 \quad 29];$$

$$H = \delta_2[1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2].$$

先找一个最小 Y 友好子空间. 考察 H, 容易得到 $n_1 = 8$, $n_2 = 24$. 于是, 我们有最大 2 型公因数为 $2^s = 2^3$, 并且 $m_1 = 1$, $m_2 = 3$. 于是, 最小 Y 友好子空间维数为 $n - s = 5 - 3 = 2$. 利用算法 12.2.1, 可得到

$$T_0 = \delta_4[1\ 2\ 3\ 4\ 1\ 2\ 3\ 4\ 1\ 2\ 3\ 4\ 1\ 2\ 3\ 4\ 2\ 1\ 4\ 3\ 2\ 1\ 4\ 3\ 2\ 1\ 4\ 3\ 2\ 1\ 4\ 3],$$

以及

$$G = \delta_2[1\ 2\ 2\ 2].$$

设最小 Y 友好子空间为 $\mathcal{Z}^2 = \mathcal{F}_\ell\{z_4, z_5\}$. 现在设

$$z_4 = M_4 x, \quad z_5 = M_5 x,$$

则不难从 T_0 算出 M_4 和 M_5 如下:

$$M_4 = \delta_2[1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2];$$
$$M_5 = \delta_2[1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1].$$

选择 $z_i = M_i x$, $i = 1, 2, 3$, 使 $z = \{z_1, z_2, z_3, z_4, z_5\}$ 为一新坐标. 这时, 只要将 M_i, $i = 1, 2, 3, 4, 5$ (略去 δ_2) 排列一矩阵

$$G := \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \end{bmatrix}.$$

如果每一列均不相同, 则 z 为一新坐标. 例如可选

$$M_1 = \delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2];$$
$$M_2 = \delta_2[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1];$$
$$M_3 = \delta_2[1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1].$$

返回逻辑形式有

$$\begin{cases} z_1 = x_1, \\ z_2 = \neg x_2, \\ z_3 = x_2 \leftrightarrow x_3, \\ z_4 = x_4, \\ z_5 = x_1 \leftrightarrow x_5. \end{cases} \tag{12.3.16}$$

记 $z = \ltimes_{i=1}^{5} z_i$ 及 $x = \ltimes_{i=1}^{5} x_i$, 那么 (12.3.16) 的代数形式为 $z = Tx$, 这里

$$
\begin{aligned}
T = \delta_{32}[&9\ 10\ 11\ 12\ 13\ 14\ 15\ 16\ 5\ 6\ 7\ 8\ 1\ 2\ 3\ 4\ 26\ 25\\
&28\ 27\ 30\ 29\ 32\ 31\ 22\ 21\ 24\ 23\ 18\ 17\ 20\ 19].
\end{aligned}
$$

反之, $x = T^{\mathrm{T}} z$, 这里

$$
\begin{aligned}
T^{\mathrm{T}} = \delta_{32}[&13\ 14\ 15\ 16\ 9\ 10\ 11\ 12\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 30\ 29\\
&32\ 31\ 26\ 25\ 28\ 27\ 18\ 17\ 20\ 19\ 22\ 21\ 24\ 23].
\end{aligned}
$$

于是, (12.3.16) 的逆变换为

$$
\begin{cases}
x_1 = z_1,\\
x_2 = \neg z_2,\\
x_3 = z_2 \bar{\vee} z_3,\\
x_4 = z_4,\\
x_5 = z_1 \leftrightarrow z_5.
\end{cases}
$$

最后, 在 z 坐标下, 系统 (12.3.14) 变为

$$
\begin{aligned}
z(t+1) &= Tx(t+1) = TLu(t)\xi(t)x(t) = TLu(t)\xi(t)T^{\mathrm{T}}z(t)\\
&= TL(I_8 \otimes T^{\mathrm{T}})u(t)\xi(t)z(t) := \tilde{L}u(t)\xi(t)z(t);
\end{aligned}
$$

及

$$
y(t) = Hx(t) = HT^{\mathrm{T}}z(t) := \tilde{H}z(t),
$$

这里

$$
\begin{aligned}
\tilde{L} = \delta_{32}[&17\ \ 17\ \ 1\ \ 1\ \ 19\ \ 19\ \ 3\ \ 3\ \ 17\ \ 17\ \ 2\ \ 2\ \ 19\ \ 19\ \ 4\ \ 4\\
&17\ \ 17\ \ 1\ \ 1\ \ 19\ \ 19\ \ 3\ \ 3\ \ 17\ \ 17\ \ 2\ \ 2\ \ 19\ \ 19\ \ 4\ \ 4\\
&17\ \ 17\ \ 1\ \ 1\ \ 27\ \ 27\ \ 11\ \ 11\ \ 19\ \ 19\ \ 4\ \ 4\ \ 27\ \ 27\ \ 12\ \ 12\\
&17\ \ 17\ \ 1\ \ 1\ \ 27\ \ 27\ \ 11\ \ 11\ \ 19\ \ 19\ \ 4\ \ 4\ \ 27\ \ 27\ \ 12\ \ 12\\
&17\ \ 21\ \ 2\ \ 6\ \ 19\ \ 23\ \ 4\ \ 8\ \ 17\ \ 21\ \ 2\ \ 6\ \ 19\ \ 23\ \ 4\ \ 8\\
&17\ \ 21\ \ 2\ \ 6\ \ 19\ \ 23\ \ 4\ \ 8\ \ 17\ \ 21\ \ 2\ \ 6\ \ 19\ \ 23\ \ 4\ \ 8\\
&17\ \ 21\ \ 2\ \ 6\ \ 27\ \ 31\ \ 12\ \ 16\ \ 19\ \ 23\ \ 4\ \ 8\ \ 27\ \ 31\ \ 12\ \ 16\\
&17\ \ 21\ \ 2\ \ 6\ \ 27\ \ 31\ \ 12\ \ 16\ \ 19\ \ 23\ \ 4\ \ 8\ \ 27\ \ 31\ \ 12\ \ 16\\
&1\ \ 1\ \ 17\ \ 17\ \ 3\ \ 3\ \ 19\ \ 19\ \ 1\ \ 1\ \ 18\ \ 18\ \ 3\ \ 3\ \ 20\ \ 20\\
&1\ \ 1\ \ 17\ \ 17\ \ 3\ \ 3\ \ 19\ \ 19\ \ 1\ \ 1\ \ 18\ \ 18\ \ 3\ \ 3\ \ 20\ \ 20\\
&1\ \ 1\ \ 17\ \ 17\ \ 11\ \ 11\ \ 27\ \ 27\ \ 1\ \ 1\ \ 18\ \ 18\ \ 11\ \ 11\ \ 28\ \ 28\\
&1\ \ 1\ \ 17\ \ 17\ \ 11\ \ 11\ \ 27\ \ 27\ \ 1\ \ 1\ \ 18\ \ 18\ \ 11\ \ 11\ \ 28\ \ 28
\end{aligned}
$$

$$
\begin{array}{cccccccccccccccc}
1 & 5 & 18 & 22 & 3 & 7 & 20 & 24 & 1 & 5 & 18 & 22 & 3 & 7 & 20 & 24 \\
1 & 5 & 18 & 22 & 3 & 7 & 20 & 24 & 1 & 5 & 18 & 22 & 3 & 7 & 20 & 24 \\
1 & 5 & 18 & 22 & 11 & 15 & 28 & 32 & 1 & 5 & 18 & 22 & 11 & 15 & 28 & 32 \\
1 & 5 & 18 & 22 & 11 & 15 & 28 & 32 & 1 & 5 & 18 & 22 & 11 & 15 & 28 & 32];
\end{array}
$$

$$\tilde{H} = \delta_2[1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2].$$

于是, 在 z 坐标下的方程变为

$$
\begin{cases}
z_1(t+1) = z_4(t)\bar{\vee}u_1(t), \\
z_2(t+1) = z_3(t) \vee \xi(t), \\
z_3(t+1) = z_5(t) \vee u_2(t), \\
z_4(t+1) = [u_1(t) \to (z_2(t) \vee \xi(t))] \wedge z_3(t), \\
z_5(t+1) = (u_2(t) \wedge z_2(t)) \vee z_4(t),
\end{cases}
\\
y = z_4 \wedge z_5. \tag{12.3.17}
$$

现在, 最小 Y 友好子空间为 (z_4, z_5). 如果选择

$$u_1(t) = z_2(t) = \neg x_2(t), \quad u_2(t) = 0.$$

那么, 只有 z_3 未消去. 但 z_3 方程中并无干扰. 于是, 只要将 (z_3, z_4, z_5) 看作 Y 友好子空间, 则显然 (12.3.17) 具有干扰解耦形式 (12.1.6).

12.4 注释与参考

干扰解耦问题 (disturbance decoupling problem (DDP)) 是一个经典的控制问题. 关于线性与非线性系统的干扰解耦问题在控制理论中均有详尽讨论, 可分别见文献 [145] 与文献 [87]. 本章讨论布尔控制网络的干扰解耦问题. 本章内容主要根据文献 [58]. 文献 [108, 148] 给出了一些新进展.

第13章　逻辑系统的稳定性与镇定

13.1　布尔矩阵的向量距离

记 $\mathcal{D}=\{1,0\}$. 首先回忆布尔矩阵/布尔向量, 再确定一个运算.

定义 13.1.1　1) 矩阵 $X=(x_{ij})\in\mathcal{M}_{m\times n}$ 称为一个布尔矩阵, 如果它的元素 $x_{ij}\in\mathcal{D}$. 当 $m=1$ $(n=1)$ 时 X 称为一个布尔行 (列) 向量. $m\times n$ 维布尔矩阵集合记作 $\mathcal{B}_{m\times n}$.

2) 设 $X=(x_{ij})\in\mathcal{B}_{m\times n}$, σ 为一元逻辑算子, 那么, $\sigma X=(\sigma x_{ij})$.

3) 设 $X=(x_{ij})$, $Y=(y_{ij})\in\mathcal{B}_{m\times n}$, σ 为二元逻辑算子, 那么 $X\sigma Y:=(x_{ij}\sigma y_{ij})$.

4) 设 $\alpha\in\mathcal{D}$, $X=(x_{ij})\in B_{m\times n}$. σ 为二元逻辑算子, 那么 $\alpha\sigma X:=(\alpha\sigma x_{ij})$.

下面是几个小例子:

例 13.1.1　1) 设

$$X=\begin{bmatrix}1&0&1\\0&1&1\end{bmatrix},$$

那么

$$\neg X=\begin{bmatrix}0&1&0\\1&0&0\end{bmatrix}.$$

2) 设 X 如上, 且

$$Y=\begin{bmatrix}1&1&0\\0&0&1\end{bmatrix},$$

那么

$$X\vee Y=\begin{bmatrix}1&1&1\\0&1&1\end{bmatrix};\quad X\bar{\vee}Y=\begin{bmatrix}0&1&1\\0&1&0\end{bmatrix}.$$

3) 设 X 及 Y 同上, $a=1$, $b=0$, 那么

$$(a\bar{\vee}X)\vee(b\leftrightarrow Y)=\begin{bmatrix}0&1&0\\1&0&0\end{bmatrix}\vee\begin{bmatrix}0&0&1\\1&1&0\end{bmatrix}=\begin{bmatrix}0&1&1\\1&1&0\end{bmatrix}.$$

下面讨论数乘与矩阵积.

定义 13.1.2 1) 设 $\alpha \in \mathcal{D}$, 那么 α 与 $X \in B_{m\times n}$ 的数乘定义为

$$\alpha X = X\alpha := \alpha \wedge X. \tag{13.1.1}$$

特别地, 当 $\alpha, \beta \in \mathcal{D}$, 则 $\alpha\beta = \alpha \wedge \beta$, 这与实数相乘一致.

2) 设 $X = (x_{ij}) \in \mathcal{B}_{p\times q}$, $Y \in \mathcal{B}_{m\times n}$, 那么, 它们的布尔 Kronecker 矩阵积定义为

$$X \otimes_{\mathcal{B}} Y = (x_{ij}Y) \in \mathcal{B}_{pm\ltimes qn}. \tag{13.1.2}$$

3) 设 $\alpha, \beta, \alpha_i \in \mathcal{D}$, $i = 1, 2, \cdots, n$. 则其布尔加法定义为

$$\begin{cases} \alpha +_{\mathcal{B}} \beta := \alpha \vee \beta, \\ \sum\limits_{\mathcal{B}}{}_{i=1}^{n} \alpha_i := \alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_n. \end{cases} \tag{13.1.3}$$

4) 设 $X = (x_{ij}) \in \mathcal{B}_{m\times n}$, $Y = (y_{ij}) \in \mathcal{B}_{n\times p}$, 那么, 它们的布尔矩阵积定义为

$$X \times_{\mathcal{B}} Y := Z \in \mathcal{B}_{m\times p}, \tag{13.1.4}$$

这里

$$z_{ij} = \sum\limits_{\mathcal{B}}{}_{k=1}^{n} x_{ik}y_{kj}, \quad i = 1, \cdots, m;\ j = 1, \cdots, p.$$

5) 设 $A \in \mathcal{B}_{m\times n}$, $B \in \mathcal{B}_{p\times q}$. 它们的布尔矩阵积定义为

$$A \times_{\mathcal{B}} B := \left(A \otimes_{\mathcal{B}} I_{t/n}\right) \times_{\mathcal{B}} \left(A \otimes_{\mathcal{B}} I_{t/p}\right), \tag{13.1.5}$$

这里 $t = \mathrm{lcm}(n, p)$ 为 n, p 的最小公倍数.

6) 设 A 为一布尔矩阵, A 的 k 次布尔幂定义为

$$A^{(k)} := \underbrace{A \times_{\mathcal{B}} A \times_{\mathcal{B}} \cdots \times_{\mathcal{B}} A}_{k}.$$

下面也给几个例子.

例 13.1.2 设

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}; \quad B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}; \quad C = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

那么

$$A +_{\mathcal{B}} B = A \vee B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}; \quad A \to B = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix};$$

$$A \leftrightarrow B = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}; \quad A \bar{\vee} B = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix};$$

$$A \times_{\mathcal{B}} C = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}; \quad B \times_{\mathcal{B}} C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

下面定义 $\mathcal{B}_{m\times n}$ 中的布尔矩阵的一些关系式.

定义 13.1.3 设 $X = (x_{ij}), Y = (y_{ij}) \in \mathcal{B}_{m\times n}$.

1) 如果 $x_{ij} \leqslant y_{ij}, \forall i, j$, 称 $X \leqslant Y$.

2) X 和 Y 的向量距离, 记作 $D_v(X, Y)$, 定义为

$$D_v(X, Y) = X \bar{\vee} Y. \tag{13.1.6}$$

布尔加法及布尔乘法都是保序的, 即如果 $a \geqslant b$ 且 $c \geqslant d$, 则

$$a +_{\mathcal{B}} c \geqslant b +_{\mathcal{B}} d;$$
$$a \times_{\mathcal{B}} c \geqslant b \times_{\mathcal{B}} d.$$

因此, 不难验证以下结论, 它们是对向量情况下的相应结果 [125] 的推广.

命题 13.1.1 设 $A \geqslant B$ 及 $C \geqslant E$, 那么

$$A \times_{\mathcal{B}} C \geqslant B \times_{\mathcal{B}} E. \tag{13.1.7}$$

命题 13.1.2 设 $X, Y, Z \in \mathcal{B}_{m\times n}$. 它们之间的向量距离满足

(i)

$$D_v(X, Y) = 0 \Rightarrow X = Y; \tag{13.1.8}$$

(ii) (对称性)

$$D_v(X, Y) = D_v(Y, X); \tag{13.1.9}$$

(iii) (三角不等式)

$$D_v(X,Z) \leqslant D_v(X,Y) +_{\mathcal{B}} D_v(Y,Z). \tag{13.1.10}$$

最后, 考虑布尔矩阵积的向量矩离.

命题 13.1.3 设 $A,\ B \in \mathcal{B}_{m\times n},\ C \in \mathcal{B}_{n\times p},\ E \in \mathcal{B}_{q\times m}$. 那么

1)

$$D_v(A \times_{\mathcal{B}} C, B \times_{\mathcal{B}} C) \leqslant D_v(A,B) \times_{\mathcal{B}} C; \tag{13.1.11}$$

2)

$$D_v(E \times_{\mathcal{B}} A, E \times_{\mathcal{B}} B) \leqslant E \times_{\mathcal{B}} D_v(A,B). \tag{13.1.12}$$

证明 只证明 (13.1.11), (13.1.12) 的证明是相同的. 根据定义, 只要证明: 对任何 $1 \leqslant i \leqslant n$ 以及 $1 \leqslant j \leqslant n$ 有

$$\sum_{\mathcal{B}}{}_{k=1}^{n} a_{ik}c_{kj} \bar{\vee} \sum_{\mathcal{B}}{}_{k=1}^{n} b_{ik}c_{kj} \leqslant \sum_{\mathcal{B}}{}_{k=1}^{n} (a_{ik}\bar{\vee} b_{ik})c_{kj}. \tag{13.1.13}$$

(13.1.13) 右边等于零, 当且仅当,

$$\text{或者} \quad a_{ik} = b_{ik} \quad \text{或者} \quad c_{kj} = 0, \quad \forall k. \tag{13.1.14}$$

但是, 不管 (13.1.14) 中哪一种情况发生, 容易检验, 此时, (13.1.13) 左边都等于零. 于是立得结论. □

注意到当 C 是个向量, 即当 $C = X \in \mathcal{B}_{n\times 1}$ 时, 不等式 (13.1.11) 变为

$$D_v(AX, BX) \leqslant D_v(A,B)X, \tag{13.1.15}$$

它在应用上特别方便.

下面的例子表明, 上述不等式有时可以成为严格不等式.

例 13.1.3 令

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix},$$

以及

$$X = \begin{bmatrix} 1 & 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}}.$$

那么

$$AX\bar{\vee}BX = \left(\begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right)\bar{\vee}\left(\begin{bmatrix}1 & 1 & 1 & 1\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right) = 1\bar{\vee}1 = 0;$$

$$(A\bar{\vee}B)X = \begin{bmatrix}0 & 1 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\0\\0\\1\end{bmatrix} = 1.$$

即

$$D_v(AX, BX) < D_v(A, B)X.$$

另一个自然的问题是: 什么时候 (13.1.11) 和 (13.1.12) 会变成等式? 实际上有如下结论, 它在后面会用到.

命题 13.1.4 设 $A, B \in \mathcal{B}_{m\times n}$, $C \in \mathcal{L}_{n\times p}$, 以及 $E \in \mathcal{L}_{q\times m}$. 那么

$$D_v(AC, BC) = D_v(A, B)C; \tag{13.1.16}$$

并且

$$D_v(EA, EB) = ED_v(A, B). \tag{13.1.17}$$

证明 只证 (13.1.16), (13.1.17) 类此可证.

由定义, 只要对 $C \in \mathcal{L}_{n\times 1}$ 证明就可以了. 因此, 不失一般性, 可设 $C = \delta_n^i$. 于是 (13.1.16) 变为

$$\mathrm{Col}_i(A)\bar{\vee}\mathrm{Col}_i(B) = \mathrm{Col}_i(A\bar{\vee}B),$$

这显然是对的. □

13.2 全局稳定性

本节讨论布尔网络的全局稳定性, 即, 是否存在一个不动点, 它是一个全局的吸引子. 换言之, 我们考虑一个布尔动态系统, 问它何时全局收敛到一个点? 本节主要利用向量距离, 得到的结果是文献 [125] 相应结果的推广.

考察一个布尔网络

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)), \\ x_2(t+1) = f_2(x_1(t), \cdots, x_n(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t)), \quad x_i \in \mathcal{D}, \end{cases} \tag{13.2.1}$$

或者一个布尔控制网络

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ x_2(t+1) = f_2(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)). \end{cases} \tag{13.2.2}$$

记 $\mathcal{X} = \mathcal{D}^n$ 为它们的状态空间. 一个点 (状态)$X \in \mathcal{X}$ 可表示为 $X = (x_1, \cdots, x_n)^{\mathrm{T}}$. 考虑一个逻辑映射 $F : \mathcal{X} \to \mathcal{X}$, 它用下述逻辑方程描述:

$$\begin{cases} z_1 = f_1(x_1, \cdots, x_n), \\ \quad\vdots \\ z_n = f_n(x_1, \cdots, x_n). \end{cases} \tag{13.2.3}$$

我们可以将它简记为

$$Z = F(X), \quad \text{这里} \quad X, Z \in \mathcal{X}. \tag{13.2.4}$$

这个映射可能来自布尔网络 (13.2.1). 即

$$X_{t+1} = F(X_t). \tag{13.2.5}$$

设映射 F 由 (13.2.3) 确定. 那么, F 的关联矩阵, 记作 $\mathcal{I}(F) := (r_{i,j}) \in \mathcal{B}_{n\times n}$, 定义为

$$r_{i,j} = \begin{cases} 1, & x_j \text{ 影响 } z_i, \\ 0, & x_j \text{ 不影响 } z_i. \end{cases}$$

定理 13.2.1 设 $X, Y \in \mathcal{X}$, $F : \mathcal{X} \to \mathcal{X}$. 那么,

$$D_v(F(X), F(Y)) \leqslant \mathcal{I}(F) \ltimes_{\mathcal{B}} D_v(X, Y), \tag{13.2.6}$$

这里 $\mathcal{I}(F)$ 是 F 的关联矩阵.

证明 记

$$\mathcal{I}(F) = (b_{ij}) \in \mathcal{B}_{n\times n}.$$

利用三角不等式

$$\begin{aligned} D_v(f_i(X), f_i(Y)) \leqslant\ & D_v(f_i(x_1,\cdots,x_n), f_i(y_1,x_2,\cdots,x_n) \\ & +D_v(f_i(y_1,x_2,\cdots,x_n), f_i(y_1,y_2,x_3,\cdots,x_n) \\ & +\cdots \\ & +D_v(f_i(y_1,\cdots,y_{n-1},x_n), f_i(y_1,\cdots,y_n) \\ \leqslant\ & \sum_{\mathcal{B}}{}_{k=1}^{n} b_{i,k} D_v(x_k,y_k). \end{aligned}$$

结论显见. □

注意: 同样的推理可知 (13.2.6) 对于 $F:\mathcal{D}^n \to \mathcal{D}^m$ 这种一般情况也是正确的.

定理 13.2.2 对于一个映射 $F:\mathcal{X}\to\mathcal{X}$, 这里 $\mathcal{X}=\mathcal{D}^n$, 如果存在一个矩阵 $M\in\mathcal{B}_{n\times n}$ 使得

$$D_v(F(X),F(Y)) \leqslant M \ltimes_{\mathcal{B}} D_v(X,Y), \quad \forall X,Y\in\mathcal{X}, \tag{13.2.7}$$

那么

$$\mathcal{I}(F) \leqslant M.$$

证明 我们用反证法证明. 设存在 M 满足 (13.2.7) 且其中有一个元素 $m_{ij} < b_{ij}$. 于是 $m_{ij}=0$ 且 $b_{ij}=1$. 由于 f_i 依赖于 x_j, 则可找到 $X=(x_1,\cdots,x_j,\cdots x_n)$ 及 $Y=(x_1,\cdots,y_j,\cdots x_n)$, 使得 $f_i(X)\neq f_i(Y)$. 这意味着,

$$D_v(f_i(X), f_i(Y)) = 1.$$

但是, 利用 (13.2.7) 有

$$D_v(f_i(X), f_i(Y)) \leqslant \sum_{\mathcal{B}}{}_{k\neq j} m_{ik} D_v(x_k,x_k) +_{\mathcal{B}} m_{ij} D_v(x_j,y_j) = 0,$$

这是荒唐的. □

定理 13.2.3 设 $E,F:\mathcal{X}\to\mathcal{X}$ 为两个逻辑映射. 那么

$$\mathcal{I}(E\circ F) \leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} \mathcal{I}(F). \tag{13.2.8}$$

证明 对任意 $X, Y \in \mathcal{X}$,

$$D_v(E \circ F(X), E \circ F(Y)) \leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} d(F(X), F(Y)) \leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} \mathcal{I}(F) \ltimes_{\mathcal{B}} D_v(x, y).$$

根据定理 13.2.2, 结论显见. □

上述定理的一个直接应用是

推论 13.2.1 设 ξ 为 (13.2.1) 的一个不动点. 那么

$$D_v(X(k), \xi) \leqslant [\mathcal{I}(F)]^{(k)} \ltimes_{\mathcal{B}} D_v(x(0), \xi). \tag{13.2.9}$$

特别是, 如果

$$\mathrm{Col}_\lambda\left([\mathcal{I}(F)]^{(k)}\right) = 0, \quad \lambda \in \Lambda := \{j_1, \cdots, j_s\} \subset \{1, 2, \cdots, n\},$$

并且

$$x_\alpha(0) = \xi_\alpha, \quad \forall \alpha \notin \{j_1, \cdots, j_s\},$$

那么 $X(t) = \xi, t \geqslant k$.

定义 13.2.1 系统 (13.2.1) 称为全局稳定的, 如果它全局收敛到一个点. 这等价于系统只有一个吸引子, 它是一个不动点.

综合以上讨论可得

命题 13.2.1 设 0 为 F 的一个不动点, 同时存在 $k > 0$ 使得

$$[\mathcal{I}(F)]^{(k)} = 0, \tag{13.2.10}$$

则 0 是一个全局吸引子. 即系统全局稳定于 0.

下面讨论一个例子.

例 13.2.1 考察如下系统

$$\begin{cases} x_1(t+1) = f_1(x_2(t), x_3(t)), \\ x_2(t+1) = f_2(x_4(t)), \\ x_3(t+1) = c_0, \\ x_4(t+1) = f_4(x_3(t)), \end{cases} \tag{13.2.11}$$

这里 f_1, f_2 和 f_3 可以是任何逻辑函数, c_0 是一个逻辑常量. F 的关联矩阵为

$$\mathcal{I}(F) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

容易检验 $[\mathcal{I}(F)]^{(4)}=0$. 如果 0 是系统 (13.2.11) 的一个不动点, 则系统全局收敛于 0.

注意到如果 $x_e=(e_1,e_2,\cdots,e_n)$ 是系统 (13.2.1) 的一个不动点, 那么, 上面的方法实际上也可以用来检验 x_e 是否是一个全局吸引子. 做一个坐标变换:

$$z_i=\begin{cases}x_i, & e_i=0,\\ \neg x_i, & e_i=1.\end{cases} \tag{13.2.12}$$

这样, 可以将 (13.2.1) 变成一个关于 z 的新系统如下:

$$z(t+1)=\tilde{F}(z(t)). \tag{13.2.13}$$

如果存在 $k>0$ 使得 $[\mathcal{I}(\tilde{F})]^{(k)}=0$, 那么 x_e 是系统 (13.2.1) 的全局吸引子.

利用以上结果, 不难验证如下结论:

命题 13.2.2　设 $H\in\mathcal{B}_{n\times n}$, 则以下几点等价:

(i) 存在一个 $k>0$ 使得 $H^{(k)}=0$;

(ii) 存在一个置换矩阵 P 使得 $P^{\mathrm{T}}\times_{\mathcal{B}}H\times_{\mathcal{B}}P$ 为一严格下三角 (或上三角) 矩阵.

实际上, 当 $H=\mathcal{I}(F)$ 为一关联矩阵时, P 表示变量的重新排列.

可惜的是, 上述结论只是充分条件而非必要条件. 见下例:

例 13.2.2　考察如下系统

$$\begin{cases}x_1(t+1)=x_1(t)\wedge x_2(t),\\ x_2(t+1)=x_1(t)\wedge(\neg x_2(t)).\end{cases} \tag{13.2.14}$$

容易检验, 0 是它的全局吸引子. 但它演化方程的关联矩阵为

$$\mathcal{I}(F)=\begin{bmatrix}1&1\\1&1\end{bmatrix},$$

并且

$$[\mathcal{I}(F)]^{(k)}=\mathcal{I}(F)\neq 0,\quad k\geqslant 1.$$

那么, 什么是全局稳定的充要条件呢? 实际上有

定理 13.2.4　布尔网络 (13.2.1) 全局收敛, 当且仅当, 存在一个 $k>0$ 使得

$$\mathcal{I}(F^k)=0. \tag{13.2.15}$$

证明 (必要性) 如果系统是全局收敛的, 那么, 经 T_t 步 (这里 T_t 是过渡时间) 所有的初始状态都收敛到一个点 ξ. 因此, 当 $k \geqslant T_t$(13.2.15) 成立.

(充分性) 设 (13.2.15) 成立. 那么对任意的 X 均有 $F^k(X)$ 为定常映射. 例如,

$$F^k(X) = \xi.$$

那么, 对任何 $t \geqslant k$,

$$F^t(X) = F^k(F^{t-k}(X)) = \xi.$$

□

注 1) 命题 13.2.2 及由其导出的方法十分有效, 因为关联矩阵的大小为 $n \times n$, 它是 $O(n)$ 阶的.

2) 在定理 13.2.3 中 F^k 不是直接可计算的. 它只能通过 F 的代数形式 L_F 来计算, 后者的大小为 $2^n \times 2^n$. 所以当 n 大的时候计算量会很大.

3) 根据定理 13.2.3 可知

$$\mathcal{I}(F^k) \leqslant [\mathcal{I}(F)]^{(k)}, \quad k \geqslant 1. \tag{13.2.16}$$

它们一般是不等的.

定义 13.2.2 设 $F : \mathcal{D}^n \to \mathcal{D}^m$. F 称为一个定常映射, 如果存在一个 $Z_0 \in \mathcal{D}^m$ 使得

$$F(X) = Z_0, \quad \forall X \in \mathcal{D}^n. \tag{13.2.17}$$

根据定义马上可推得如下结果:

命题 13.2.3 1) F 为一定常映射, 当且仅当, 存在定常逻辑变量 $z_0 \in \Delta_{2^m}$, 使其结构矩阵 M_F 满足

$$\mathrm{Col}_i(M_F) = z_0, \quad 1 \leqslant i \leqslant 2^n. \tag{13.2.18}$$

2) $\mathcal{I}(F) = 0$, 当且仅当 F 是一个定常映射.

回忆命题 13.2.2. 实际上 0 是一个不动点不是稳定的一个必要条件. 因为我们将网络状态空间视为具有离散拓扑的拓扑空间, 稳定就意味着全局收敛. 因为由 (13.2.16) 容易看出, 条件 (13.2.10) 保证了 F^s 当 $s \geqslant k$ 时是定常的. 例如, $F^s(X) = \xi, \forall X$ 以及 $s \geqslant k$. 那于, 系统就会全局收敛于 ξ. 我们将这一点写成一个推论.

推论 13.2.2 考察系统 (13.2.1). 如果条件 (13.2.10) 成立, 则它是全局稳定的.

命题 13.2.2 是稳定性分析与镇定的一个基本工具, 因此, 下面对它做一点进一步的讨论.

首先, 必须指出, 一个布尔网络, 其动态映射 F 的关联矩阵 $\mathcal{I}(F)$ 是与坐标变换相关的. 例如, 见下面的例子.

例 13.2.3 考察如下系统

$$\begin{cases} x_1(t+1) = [x_1(t) \wedge (x_2(t) \bar{\vee} x_3(x))] \vee (\neg x_1(t) \wedge x_3(t)), \\ x_2(t+1) = [x_1(t) \wedge (\neg x_2(t))] \vee (\neg x_1 \wedge x_2), \\ x_3(t+1) = [x_1(t) \wedge (\neg(x_2(t) \wedge x_3(t)))] \vee [\neg x_1(t) \wedge (x_2(t) \vee x_3(t))]. \end{cases} \tag{13.2.19}$$

简记它为

$$x(t+1) = F(x(t)).$$

容易检验 0 是 (13.2.19) 的一个不动点. 这个系统的关联矩阵为

$$\mathcal{I}(F) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

无论怎样改变状态变量的顺序都不可能将其变为严格下三角的形式. 在代数形式下 (13.2.19) 可表示为

$$x(t+1) = Lx(t), \tag{13.2.20}$$

这里 $x(t) = x_1(t)x_2(t)x_3(t)$,

$$L = \delta_8[8\ 3\ 1\ 5\ 1\ 5\ 3\ 8].$$

考虑如下的坐标变换

$$\begin{cases} z_1 = [x_1 \wedge \neg(x_3)] \vee [(\neg x_1) \wedge (x_2 \bar{\vee} x_3)], \\ z_2 = [x_1 \wedge (x_2 \bar{\vee} x_3)] \vee [(\neg x_1) \wedge x_3], \\ z_3 = x_2. \end{cases} \tag{13.2.21}$$

在向量形式下, 容易算得

$$z = z_1 z_2 z_3 = Tx,$$

这里

$$T = \delta_8[7\ 1\ 6\ 4\ 5\ 3\ 2\ 8].$$

那么, 在 z 坐标下有

$$z(t+1) = TLT^{\mathrm{T}} z(t) := \tilde{L} z(t), \tag{13.2.22}$$

这里 $\tilde{L}$ 为

$$\tilde{L} = \delta_8[6\ 6\ 5\ 5\ 7\ 7\ 8\ 8].$$

将代数表达式返回逻辑表达式的标准方法见文献 [55], 简述如下: 构造一组 2×2^n 矩阵, 称为重构矩阵, 如下:

$$S_k^n = \delta_2[\underbrace{1 \cdots 1}_{2^{n-k}}\underbrace{2 \cdots 2}_{2^{n-k}} \cdots \underbrace{1 \cdots 1}_{2^{n-k}}\underbrace{2 \cdots 2}_{2^{n-k}}], \quad k = 1, \cdots, n. \tag{13.2.23}$$

利用重构矩阵, 可以由 $\tilde{L}$ 得到每个逻辑变量的动态方程.

设其动态方程的代数表达式为

$$\begin{cases} z_1(t+1) = M_1 z(t), \\ z_2(t+1) = M_2 z(t), \\ z_3(t+1) = M_3 z(t), \end{cases} \tag{13.2.24}$$

则

$$\begin{aligned} M_1 &= S_1^3 \tilde{L} = \delta_2[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2]; \\ M_2 &= S_2^3 \tilde{L} = \delta_2[1\ 1\ 1\ 1\ 2\ 2\ 2\ 2]; \\ M_3 &= S_3^3 \tilde{L} = \delta_2[2\ 2\ 1\ 1\ 1\ 1\ 2\ 2]. \end{aligned}$$

再用定理 3.4.1 的证明中用到的方法即可将 (13.2.24) 返回逻辑形式. 略去过程, 最后可得

$$\begin{cases} z_1(t+1) = 0, \\ z_2(t+1) = z_1(t), \\ z_3(t+1) = z_1(t) \bar{\vee} z_2(t). \end{cases} \tag{13.2.25}$$

考察系统 (13.2.25)(即系统 (13.2.19) 在 Z 坐标下的表现), 其关联矩阵为

$$\mathcal{I}(\tilde{F}) = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix},$$

它是严格下三角的. 由于 $x_1 = x_2 = x_3 = 0$ 是 (13.2.19) 的一个不动点, 我们可知 (13.2.19) 全局收敛于 0.

例 13.2.3 表明, 有时坐标变换可以帮助我们找到合适的关联矩阵, 从而判定出全局收敛性. 那么, 一个自然的问题就是: 对于一个全局稳定的网络, 是否一定有

一个合适的坐标, 使其演化方程具有严格下 (或上) 三角的关联矩阵? 遗憾的是, 回答是否定的. 考察例子 13.2.2. 因为 $n=2$ 这里共有 $2^2!=24$ 种坐标变换. 它们是

$$T_1=I_2;\quad T_2=\delta_4[1\ 2\ 4\ 3];\quad T_3=\delta_4[1\ 3\ 2\ 4];\cdots;T_{24}=\delta_4[4\ 3\ 2\ 1].$$

直接计算可得

$$T_2:\begin{cases}z_1=x_1,\\ z_2=x_1\leftrightarrow x_2.\end{cases}\quad\Rightarrow\quad\begin{cases}z_1(t+1)=z_1(t)\wedge z_2(t),\\ z_2(t+1)=\neg z_1(t).\end{cases}$$

$$T_3:\begin{cases}z_1=x_2,\\ z_2=x_1\leftrightarrow x_2.\end{cases}\quad\Rightarrow\quad\begin{cases}z_1(t+1)=z_1(t)\wedge z_2(t),\\ z_2(t+1)=(\neg z_1(t))\wedge z_2(t).\end{cases}$$

$\cdots$

$$T_{24}:\begin{cases}z_1=\neg x_1,\\ z_2=x_2.\end{cases}\quad\Rightarrow\quad\begin{cases}z_1(t+1)=z_1(t)\wedge z_2(t),\\ z_2(t+1)=z_2(t)\to z_1(t).\end{cases}$$

我们有这么多不同表现形式, 但是, 没有一个具有严格下 (或上) 三角的关联矩阵. 因此, 命题 13.2.2 的条件对所有可能的坐标变换都不满足. 我们只能求助于定理 13.2.4.

13.3　布尔控制网络的镇定

考察布尔控制网络 (13.2.2), 记 $x(t)=\ltimes_{i=1}^{n}x_i(t)$ 及 $u(t)=\ltimes_{i=1}^{m}u_i(t)$.

定义 13.3.1　*系统 (13.2.2) 的全局镇定问题为: 找到 $u(t)$ 使得系统 (13.2.2) 全局收敛. 如果控制为如下镇定函数 $u(t)=Wx(t)$, 则称为状态反馈控制.*

命题 13.2.2 及其相关讨论是本节的主要工具.

首先给一个简单例子.

例 13.3.1　考察如下系统

$$\begin{cases}x_1(t+1)=x_3(t)\vee u(t),\\ x_2(t+1)=\neg x_1(t),\\ x_3(t+1)=x_1(t)\leftrightarrow x_2(t),\end{cases}\tag{13.3.1}$$

显然, 如果我们能用控制 $u(t)$ 将 $x_3(t)$ 消去, 则系统变为全局稳定, 因为其关联矩阵变为严下三角阵. 这一点很容易做到. 一种方法是采用开环控制 $u(t)=1$, 另一种方法是采用闭环 (状态反馈) 控制 $u(t)=\neg x_3(t)$.

为了得到一个一般的设计方法, 首先回顾逻辑状态变量的表达. 设 $x_1, \cdots, x_n$ 为 n 个逻辑状态变量. 在标量表达下有 $x_i \in \mathcal{D}$, $i = 1, \cdots, n$, 在向量表达下 $x = \ltimes_{i=1}^n x_i \in \Delta_{2^n}$.

定义一组向量:

$$s_k^n = [\underbrace{1 \cdots 1}_{2^{n-k}} \underbrace{0 \cdots 0}_{2^{n-k}} \cdots \underbrace{1 \cdots 1}_{2^{n-k}} \underbrace{0 \cdots 0}_{2^{n-k}}], \quad k = 1, \cdots, n. \tag{13.3.2}$$

注 1) $s_k^n \in \mathcal{B}_{1 \times 2^n}$, $k = 1, \cdots, n$. 因此, 逻辑算子可以在此上定义.

2) 比较 (13.3.2) 与 (13.2.23), 易知 s_k^n 可由 S_k^n 通过将括号内的 2 代替为 0 而得到.

然后, 可构造矩阵

$$\mathcal{S}^n = \begin{bmatrix} s_1^n \\ s_2^n \\ \vdots \\ s_n^n \end{bmatrix} \in \mathcal{B}_{n \times 2^n}.$$

不难验证如下命题:

命题 13.3.1 1) 从数量形式到向量形式, 有

$$x = [(x_1 \leftrightarrow s_1^n) \wedge (x_2 \leftrightarrow s_2^n) \wedge \cdots \wedge (x_n \leftrightarrow s_n^n)]^{\mathrm{T}}, \quad \forall x_i \in \mathcal{D}. \tag{13.3.3}$$

2) 从向量形式到数量形式, 有

$$X = \mathcal{S}^n x. \tag{13.3.4}$$

例 13.3.2 设 $X = (1, 0, 1, 0)^{\mathrm{T}}$. 那么, 在向量形式有

$$\begin{aligned} x &= [1 \leftrightarrow (1111111100000000)] \wedge [0 \leftrightarrow (1111000011110000)] \\ &\quad \wedge [1 \leftrightarrow (1100110011001100)] \wedge [0 \leftrightarrow (1010101010101010)] \\ &= (1111111100000000) \wedge (0000111100001111) \\ &\quad \wedge (1100110011001100) \wedge (0101010101010101) \\ &= (0000010000000000) = \delta_{16}^6. \end{aligned}$$

设 $x = \delta_{16}^9$. 那么, 在数量形式有

$$X = \mathcal{S}^4 x = (0, 1, 1, 1).$$

下面讨论布尔控制网络镇定器的一般设计方法.

首先, 设计一个映射 $\pi : \mathcal{B}_{2^n \times 2^n} \to \mathcal{B}_{n \times n}$ 如下

$$\pi(L) = [[(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L M_n)] \ltimes_{\mathcal{B}} \mathbf{1}_{2^n}, [(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L)(I_2 \otimes M_n)] \ltimes_{\mathcal{B}} \mathbf{1}_{2^n}, \cdots, \\ [(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L)(I_{2^{n-1}} \otimes M_n)] \ltimes_{\mathcal{B}} \mathbf{1}_{2^n} \ltimes_{\mathcal{B}} \mathbf{1}_{2^n}], \quad L \in \mathcal{B}_{2^n \times 2^n}. \tag{13.3.5}$$

这里 M_n 是逻辑算子 "非" 的结构矩阵.

利用映射 π, 下面这个定理告诉我们, 如何由系统转移矩阵 L 构造出邻接矩阵.

定理 13.3.1 考察布尔网络 (13.2.1) (或, 等价地, 看 (13.2.5)). 设其代数状态空间表示式为

$$x(t+1) = Lx(t), \quad x(t) \in \Delta_{2^n}, \tag{13.3.6}$$

这里 $L \in \mathcal{L}_{2^n \times 2^n}$. 那么, 其邻接矩阵 F 可通过以下公式由 L 得出

$$\mathcal{I}(F) = \pi(L). \tag{13.3.7}$$

证明 由 $\mathcal{S}^n$ 的构造容易看出 $L_s := \mathcal{S}^n L$ 是 $F : \Delta_{2^n} \to \mathcal{D}^n$ 的结构矩阵. 而当 x_1 用 $\neg x_1$ 代替时, 结构矩阵变为 $L_s M_n$. 而这时如果第 i 行不改变, 则说明 f_i 与 x_1 无关. 于是此时 $[(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L M_n)]$ 的第 i 行恒为 0. 因此 $[(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L M_n)] \ltimes_{\mathcal{B}} \mathbf{1}_{2^n} \ltimes_{\mathcal{B}} \mathbf{1}_{2^n}$ 的第 i 个元素为 0. 否则, 如果上式第 i 行至少有一个元素为 1, 则 $[(\mathcal{S}^n L)\bar{\nabla}(\mathcal{S}^n L M_n)] \ltimes_{\mathcal{B}} \mathbf{1}_{2^n} \ltimes_{\mathcal{B}} \mathbf{1}_{2^n}$ 的第 i 个元素为 1.

同样的讨论也可用于其他变量. 唯一的不同是: 必须把 "非" 算符 $\neg$ 从 x_i 前移到所有变量前. 这样就产生了 (13.3.7). □

我们给一个例子.

例 13.3.3 设系统 (13.2.1) 的传递矩阵 L 为

$$L = \delta_{16}[1\ 9\ 9\ 13\ 4\ 12\ 12\ 16\ 2\ 1\ 10\ 14\ 1\ 9\ 9\ 13]. \tag{13.3.8}$$

直接计算即可得到

$$\mathcal{I}(F) = \pi(L) = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}.$$

实际上, 利用 L 不难将逻辑系统还原出来:

$$\begin{cases} x_1(t+1) = x_3(t) \wedge x_4(t), \\ x_2(t+1) = x_3(t) \vee x_4(t), \\ x_3(t+1) = x_1(t) \to x_2(t), \\ x_4(t+1) = x_1(t) \leftrightarrow x_2(t). \end{cases} \tag{13.3.9}$$

由此可检验所得 $\mathcal{I}(F)$ 是正确的.

下面讨论镇定问题. 考察系统 (13.2.2), 其代数表达式为

$$x(t+1) = Lu(t)x(t), \quad x(t) \in \Delta_{2^n},\ u(t) \in \Delta_{2^m}, \tag{13.3.10}$$

这里 $L \in \mathcal{L}_{2^n \times 2^{n+m}}$.

根据命题 13.2.1 和 13.2.2, 可以得到利用开环控制镇定布尔网络的一个充分条件:

引理 13.3.1 系统 (13.2.2) 可以被开环控制 u 所镇定, 如果 $\pi(L(u))$ 具有严格上 (下) 三角形式.

注意到关联矩阵是依赖于坐标的, 因此, 适当的坐标变换有助实现镇定.

利用公式 (10.2.12)–(10.2.13), 可得如下结论:

定理 13.3.2 考虑系统 (13.2.2), 其代数状态空间表达式为 (13.3.10). 该系统可被开环控制 u 镇定, 如果存在坐标变换 $z = Tx$ 使得 $\pi(TL(I_{2^m} \otimes T^{\mathrm{T}})u)$ 具有严格上 (下) 三角结构.

注意到可能的控制值 u 和坐标变换都是有限的 (准确地说, 分别为 2^m 和 $2^n!$), 因此, 引理 13.3.1 和定理 13.3.2 均有限可检验的.

例 13.3.4 考察如下系统:

$$\begin{cases} x_1(t+1) = \neg x_2(t), \\ x_2(t+1) = \neg x_4(t) \leftrightarrow ((x_4(t) \wedge (x_2(t) \bar{\vee} x_3(t))) \vee u(t)), \\ x_3(t+1) = \neg((x_4(t) \wedge (x_2(t) \bar{\vee} x_3(t))) \vee u(t)), \\ x_4(t+1) = (x_4(t) \vee (x_2(t) \bar{\vee} x_3(t))) \wedge u(t). \end{cases} \tag{13.3.11}$$

通过尝试, 选用如下坐标变换:

$$\begin{cases} z_1 = x_4, \\ z_2 = x_2 \bar{\vee} x_3, \\ z_3 = \neg x_3, \\ z_4 = \neg x_1, \end{cases} \tag{13.3.12}$$

其逆变换为

$$\begin{cases} x_1 = \neg z_4, \\ x_2 = z_2 \leftrightarrow z_3, \\ x_3 = \neg z_3, \\ x_4 = z_1, \end{cases} \tag{13.3.13}$$

在新坐标下系统变为

$$\begin{cases} z_1(t+1) = (z_1(t) \vee z_2(t)) \wedge u(t), \\ z_2(t+1) = \neg z_1(t), \\ z_3(t+1) = (z_1(t) \wedge z_2(t)) \vee u(t), \\ z_4(t+1) = z_2(t) \leftrightarrow z_3(t). \end{cases} \tag{13.3.14}$$

显然, 如果设控制

$$u(t) = 0, \tag{13.3.15}$$

系统变为

$$\begin{cases} z_1(t+1) = 0, \\ z_2(t+1) = \neg z_1(t), \\ z_3(t+1) = z_1(t) \wedge z_2(t), \\ z_4(t+1) = z_2(t) \leftrightarrow z_3(t). \end{cases} \tag{13.3.16}$$

不难看出, 系统 (13.3.16) 的关联矩阵为

$$\mathcal{I}(F) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix},$$

它是严下三角的. 因此, 定常控制 $u(t) = 0$ 可镇定系统 (13.3.11).

下面考虑闭环控制. 设 $u(t)$ 为 $x(t)$ 的逻辑函数. 则可将其表示为

$$u(t) = Gx(t), \tag{13.3.17}$$

这里 $G \in \mathcal{L}_{2^m \times 2^n}$. 代入 (13.3.10) 则得

$$x(t+1) = LGx^2(t) = LGM_{r,2^n}x(t), \tag{13.3.18}$$

这里 $M_{r,s}$ 为降阶矩阵.

以下的结果是显然的.

定理 13.3.3 系统 (13.2.2) 可由闭环控制 $u = Gx$ 镇定, 如果 $\pi(LGM_{r,2^n})$ 具有严格上 (下) 三角结构. 另外, 如果存在一个坐标变换 $z = Tx$, 使得 $\pi(TLGM_{r,2^n}T^{\mathrm{T}})$ 具有严格上 (下) 三角结构, 则该控制为可镇定系统.

例 13.3.5 考察如下系统

$$\begin{cases} x_1(t+1) = \neg x_2(t), \\ x_2(t+1) = \neg x_4(t) \leftrightarrow ((x_4(t) \wedge (x_2(t) \bar{\vee} x_3(t))) \vee u(t)), \\ x_3(t+1) = \neg((x_4(t) \wedge (x_2(t) \bar{\vee} x_3(t))) \vee u(t)), \\ x_4(t+1) = (x_4(t) \vee (x_2(t) \bar{\vee} x_3(t))) \vee u(t). \end{cases} \tag{13.3.19}$$

实际上它是由 (13.3.11) 通过改变输入形式而得到的. 利用与例 13.3.4 相同的坐标变换, 我们可得

$$\begin{cases} z_1(t+1) = (z_1(t) \vee z_2(t)) \vee u(t), \\ z_2(t+1) = \neg z_1(t), \\ z_3(t+1) = (z_1(t) \wedge z_2(t)) \wedge u(t), \\ z_4(t+1) = z_2(t) \leftrightarrow z_3(t). \end{cases} \tag{13.3.20}$$

不难验证, 定常 (即开环) 控制不能镇定这个系统. 利用闭环控制

$$u(t) = \neg z_1(t) \wedge \neg z_2(t),$$

则系统 (13.3.20) 变为

$$\begin{cases} z_1(t+1) = 1, \\ z_2(t+1) = \neg z_1(t), \\ z_3(t+1) = 0, \\ z_4(t+1) = z_2(t) \leftrightarrow z_3(t), \end{cases} \tag{13.3.21}$$

显然它是全局稳定的. 将控制变回原坐标可知:

$$u(t) = \neg x_4(t) \wedge \neg(x_2(t) \bar{\vee} x_3(t)),$$

它能镇定系统 (13.3.19).

在稳定性与镇定问题的讨论中, 利用基于向量距离的方法, 其好处是所涉及的矩阵维数小, 便于计算. 其弱点在于只能得到一些充分条件. 下面寻找充要条件.

由上一节研究可知, 布尔网络的全局稳定性等价于存在整数 $k > 0$ 使 F^k 为定常映射. 在代数形式下有: L^k 所有的列均相等, 这个值就是全局吸引子. 先考虑定常控制 u 下的镇定. 此时, 控制系统的转移矩阵为 Lu.

利用矩阵半张量积性质, 容易得到

$$(Lu)^k = L[(I_{2^m} \otimes L)M_{r,2^m}]^{m-1}u. \tag{13.3.22}$$

由于只需考虑 k 小于或等于过渡时间, 即 $k \leqslant T_t \leqslant 2^n$, 我们可以如下处理: 注意到 $L[(I_{2^m} \otimes L)M_{r,2^m}]^{m-1}$ 是一个 $2^n \times 2^{n+m}$ 矩阵, 将其分为 2^m 个方阵如下,

$$L[(I_{2^m} \otimes L)\Phi_m]^{m-1} := [L_1^k \ L_2^k \ \cdots \ L_{2^m}^k]. \tag{13.3.23}$$

利用这个记号并结合以上讨论, 我们可以得到如下的易于检验的充要条件:

定理 13.3.4 系统 (13.2.2) 可由定常控制 u 镇定, 当且仅当, 在分割 (13.3.23) 中存在一个定常映射方阵

$$L_j^k, \quad 1 \leqslant k \leqslant 2^n, \ 1 \leqslant j \leqslant 2^m.$$

而且, 相应于这个定常映射方阵 L_j^k, 镇定控制为 $u = \delta_{2^m}^j$.

下面用一个例子来描述这个结果.

例 13.3.6 考察如下系统:

$$\begin{cases} x_1(t+1) = (x_1(t) \vee x_2(t)) \wedge u \\ x_2(t+1) = (x_2(t) \wedge u) \to x_1. \end{cases} \tag{13.3.24}$$

容易算得

$$L = \delta_4[1\ 1\ 2\ 3\ 3\ 3\ 3\ 3].$$

由于 $M_{r,2} = \delta_4[1, 4]$, 根据定理 13.3.4 应计算

$$L[(I_2 \otimes L)M_{r,2}]^k, \quad k \geqslant 1$$

以检验是否能找到定常映射矩阵. 实际上, 当 $k = 2$ 有

$$L[(I_2 \otimes L)M_r]^2 = \delta_4[1\ 1\ 1\ 1\ 3\ 3\ 3\ 3].$$

于是可知, 利用 $u = 1$ 可将系统镇定到 $x = \delta_4^1$(即, $x_1 = 1$, $x_2 = 1$); 而 $u = 0$ 可将系统镇定到 $x = \delta_4^3$(即, $x_1 = 0$, $x_2 = 1$).

考虑形如 (13.3.17) 的状态反馈控制. 利用表达式 (13.3.18) 以及上面的讨论, 不难得到如下结论:

定理 13.3.5 系统 (13.2.2) 可由闭环控制 $u = Gx$ 所镇定, 当且仅当, 存在一个 $2^m \times 2^n$ 的逻辑矩阵 G 和一个整数 $1 \leqslant k \leqslant 2^n$ 使得 $(LGM_{r,2^n})^k$ 是一个定常映射矩阵.

下面给出一个例子.

例 13.3.7 考虑如下系统

$$
\begin{cases}
x_1(t+1) = [x_2(t) \vee (\neg x_2(t) \wedge (x_3(t) \vee x_4(t)))] \wedge u, \\
x_2(t+1) = (x_2(t) \wedge (x_3(t) \vee x_4(t))) \vee [x_1(t) \wedge (\neg x_2(t) \\
\qquad \wedge \neg (x_3(t) \vee x_4(t)))], \\
x_3(t+1) = (x_1(t) \wedge (x_3(t) \leftrightarrow x_4(t))) \vee [\neg x_1(t) \wedge ((x_2(t) \\
\qquad \wedge (x_3(t) \leftrightarrow x_4(t))) \wedge (\neg x_2(t) \wedge (x_3(t) \wedge x_4(t))))], \\
x_4(t+1) = x_1(t) \wedge \neg x_4(t) \vee [\neg x_1(t) \wedge ((x_2(t) \wedge \neg x_4(t)) \\
\qquad \vee (\neg x_2(t) \wedge \neg (x_3(t) \rightarrow x_4(t))))].
\end{cases}
\tag{13.3.25}
$$

容易检验, 如果选择

$$
G = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}^3,
$$

那么

$$
(LGM_{r,16})^{14} = \delta_{16}[\underbrace{16 \ \cdots \ 16}_{16}].
$$

注意到

$$
u(t) = Gx(t) = x_1(t),
$$

它将系统 (13.3.25) 全局镇定到 $x_1 = 0$, $x_2 = 0$, $x_3 = 0$, $x_4 = 0$.

下面简单讨论让系统全局收敛到一个特定状态 x_0 的镇定问题. 其实这不比普通镇定问题更复杂. 除了以上的镇定条件外, 我们还要求 x_0 是控制系统的一个不动点就够了. 我们将其写成一个推论形式:

推论 13.3.1 1) 系统 (13.2.2) 可由一定常控制 u 全局镇定到 x_0, 当且仅当

$$
Lux_0 = x_0, \tag{13.3.26}
$$

并且存在一个整数 $k > 0$, 使得 $(Lu)^k$ 为一定常映射矩阵.

2) 系统 (13.2.2) 可由一闭环控制 $u = Gx$ 全局镇定到 x_0, 当且仅当, G 满足

$$
LGM_{r,2^n}x_0 = x_0, \tag{13.3.27}
$$

并且存在一个整数 $k > 0$, 使得 $(LGM_{r,2^n})^k$ 为一定常映射矩阵.

最后, 考虑利用开环控制序列 $u(t) = \ltimes_{i=1}^m u_i(t)$, $t = 1, 2, \cdots$ 的镇定问题. 设我们的目标是将系统全局镇定到 x_0.

首先, 显然必须有一个控制 $u_e \in \Delta_{2^m}$ 使得

$$Lu_e x_0 = x_0. \tag{13.3.28}$$

其次, 注意到

$$x(t_0+k+1) = Lu(t_0+k)Lu(t_0+k-1)\cdots Lu(t_0+1)x(t_0).$$

要使所有的轨道都收敛到 x_0, 必然存在一个整数 $k>0$ 使得

$$Lu(k)Lu(k-1)\cdots Lu(1)x \equiv x_0, \quad \forall x \in \Delta_{2^n}.$$

这等价于

$$\mathrm{Col}(Lu(k)Lu(k-1)\cdots Lu(1)) = \{x_0\}. \tag{13.3.29}$$

观察到

$$\begin{aligned}
&Lu(k)Lu(k-1)\cdots Lu(1)\\
&= L(I_{2^m}\otimes L)(I_{2^{2m}}\otimes L)\cdots(I_{2^{(k-1)m}}\otimes L)\ltimes_{i=k}^{1} u(i)\\
&:= [L_1^k, L_2^k, \cdots, L_{2^{km}}^k]\ltimes_{i=k}^{1} u(i).
\end{aligned} \tag{13.3.30}$$

因此, 显然存在一个 $1 \leqslant j \leqslant 2^{km}$ 使得 L_j^k 对应于一个定常映射 $\psi(x) \equiv x_0$, 而我们可以选择一个控制序列

$$\ltimes_{i=k}^{1} u(i) = \delta_{2^{mn}}^{j},$$

使 (13.3.29) 成立.

总结以上的讨论, 可得出

定理 13.3.6 系统 (13.2.2) 可由控制序列 $u(t)$, $t=1,2,\cdots$ 全局镇定到 x_0, 当且仅当,

(i) 存在一个整数 $k>0$ 和一个方阵 L_j^k, $1 \leqslant j \leqslant 2^{km}$, 使得

$$\mathrm{Col}(L_j^k) = \{x_0\};$$

(ii) 存在一个 $u_e \in \Delta_{2^m}$ 使得 (13.3.28) 成立.

下面给一个例子.

例 13.3.8 考虑如下系统

$$\begin{cases}
x_1(t+1) = x_1(t) \vee u_1(t),\\
x_2(t+1) = (x_2(t) \vee x_3(t)) \leftrightarrow u_1(t),\\
x_3(t+1) = (u_1(t) \to x_2(t)) \vee x_3(t),\\
x_4(t+1) = (x_3(t) \wedge u_2(t)) \to x_4(t).
\end{cases} \tag{13.3.31}$$

令 $x(t) = \ltimes_{i=1}^4 x_i(t)$, $u(t) = \ltimes_{i=1}^2 u_i(t)$. 表示成代数状态空间形式, (13.3.31) 变成

$$x(t+1) = Lu(t)x(t), \tag{13.3.32}$$

这里

$$\begin{aligned} L = \delta_{16}[&1\ 2\ 1\ 1\ 1\ 2\ 7\ 7\ 1\ 2\ 1\ 1\ 1\ 2\ 7\ 7\\ &1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\ 1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\\ &5\ 6\ 5\ 5\ 5\ 6\ 1\ 1\ 13\ 14\ 13\ 13\ 13\ 14\ 9\ 9\\ &5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\ 13\ 13\ 13\ 13\ 13\ 13\ 9\ 9]. \end{aligned}$$

利用 (13.3.30), 计算

$$L(I_{2^2} \otimes L)(I_{2^{2\times 2}} \otimes L) \cdots (I_{2^{2(k-1)}} \otimes L)$$

以检验是否有定常映射块. 实际上, 当 $k=2$ 有

$$M = L(I_{2^2} \otimes L) := [M_1, M_2, \cdots, M_{16}],$$

这里

$$\begin{aligned} M = \delta_{16}[&1\ 2\ 1\ 1\ 1\ 2\ 7\ 7\ 1\ 2\ 1\ 1\ 1\ 2\ 7\ 7\\ &1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\ 1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\\ &1\ 2\ 1\ 1\ 1\ 2\ 1\ 1\ 1\ 2\ 1\ 1\ 1\ 2\ 1\ 1\\ &1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\\ &1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\ 1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\\ &1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\ 1\ 1\ 1\ 1\ 1\ 1\ 7\ 7\\ &1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\\ &1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\\ &5\ 6\ 5\ 5\ 5\ 6\ 1\ 1\ 5\ 6\ 5\ 5\ 5\ 6\ 1\ 1\\ &5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\ 5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\\ &5\ 6\ 5\ 5\ 5\ 6\ 5\ 5\ 13\ 14\ 13\ 13\ 13\ 14\ 13\ 13\\ &5\ 5\ 5\ 5\ 5\ 5\ 5\ 5\ 13\ 13\ 13\ 13\ 13\ 13\ 13\ 13\\ &5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\ 5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\\ &5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\ 5\ 5\ 5\ 5\ 5\ 5\ 1\ 1\\ &5\ 5\ 5\ 5\ 5\ 5\ 5\ 5\ 13\ 13\ 13\ 13\ 13\ 13\ 13\ 13\\ &5\ 5\ 5\ 5\ 5\ 5\ 5\ 5\ 13\ 13\ 13\ 13\ 13\ 13\ 13\ 13], \end{aligned} \tag{13.3.33}$$

并且, $M_i \in \mathcal{L}_{16\times 16}, i = 1, \cdots, 16$. 由 (13.3.33) 可知

$$M_4 = M_7 = M_8 = \delta_{16}[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1].$$

也就是说,

$$\mathrm{Col}(M_4) = \mathrm{Col}(M_7) = \mathrm{Col}(M_8) = \{\delta_{16}^1\}.$$

另一方面, 选 $u_e = \delta_4^1 \sim (1,1)$(或 $u_e = \delta_4^2 \sim (1,0)$), 有

$$Lu_e\delta_{16}^1 = \delta_{16}^1.$$

根据定理 13.3.6, 利用开环控制 $u(t)$(或 $\bar{u}(t)$, 或 $\tilde{u}(t)$), 系统 (13.3.31) 可全局镇定到 $x_0 = \delta_{16}^1 \sim (1,1,1,1)$, 这里

$$u(t) = \begin{cases} \delta_4^4 \sim (0,0), & t=1, \\ \delta_4^1 \sim (1,1), & t=2, \\ u_e, & t \geqslant 3, \end{cases}$$

$$\bar{u}(t) = \begin{cases} \delta_4^3 \sim (0,1), & t=1, \\ \delta_4^2 \sim (1,0), & t=2, \\ u_e, & t \geqslant 3, \end{cases}$$

$$\tilde{u}(t) = \begin{cases} \delta_4^4 \sim (0,0), & t=1, \\ \delta_4^2 \sim (1,0), & t=2, \\ u_e, & t \geqslant 3. \end{cases}$$

下面对定理 13.3.6 中的条件做一点进一步的讨论. 条件 (i) 表示所有的轨线都可以到达预先指定的状态 x_0, 这很合理. 有人或许会怀疑条件 (ii) 是否必要. 实际上, 条件 (ii) 说的是 x_0 是某个控制下的不动点. 下面的例子说明, 条件 (ii) 也是必要的.

例 13.3.9　考虑如下系统

$$\begin{cases} x_1(t+1) = \neg(x_1(t) \wedge u(t)), \\ x_2(t+1) = (u(t) \wedge (x_2(t) \rightarrow x_1(t))) \vee (\neg u(t) \wedge (x_1(t) \leftrightarrow x_2(t))). \end{cases} \tag{13.3.34}$$

令 $x(t) = \ltimes_{i=1}^2 x_i(t)$. 利用代数状态空间表示, (13.3.34) 变为

$$\begin{aligned} x(t+1) &= Lu(t)x(t) \\ &= \delta_4[3\ 3\ 2\ 1\ 1\ 2\ 2\ 1]u(t)x(t). \end{aligned} \tag{13.3.35}$$

对于任何初值 $\xi \in \Delta_4$, 如果选择 $u(1) = \delta_2^2$, 那么

$$\begin{aligned} x(2) &= Lu(1)\xi \\ &= \delta_4[1\ 2\ 2\ 1]\xi. \end{aligned}$$

其次, 选择 $u(2)=\delta_2^1$, 那么

$$\begin{aligned}x(3)&=Lu(2)x(2)\\&=(\delta_4[3\ 3\ 2\ 1])(\delta_4[1\ 2\ 2\ 1])\xi\\&=\delta_4[3\ 3\ 3\ 3]\xi\\&=\delta_4^3,\quad \forall\xi\in\Delta_4.\end{aligned}$$

由 (13.3.35) 不难看出, 不管选什么 u_e, 都有

$$Lu_e\delta_4^3\neq\delta_4^3.$$

于是在第 3 步, 不管怎么选控制 $u(3)(u(3)=\delta_2^1$ 或 $u(3)=\delta_2^2)$, 状态轨线都会离开 $x_0=\delta_4^3$. 因此系统 (13.3.34) 不可能被任何开环控制序列全局镇定到 x_0.

13.4 注释与参考

稳定性和镇定是动态系统及其相关的动态控制系统必须讨论的最重要问题之一, 布尔 (控制) 网络当然不会是例外. 文献 [125] 系统地讨论了利用向量距离确定布尔映射的收敛性问题, 本章的部分结果是基于他们的方法. 但他们的方法只能给出充分性结果. 要得到充要条件还得回到半张量积和布尔 (控制) 系统的代数状态空间表示方法. 本章的主要结果来自文献 [59].

本章仅讨论布尔 (控制) 网络. 不过, 本章讨论的方法不难推广到一般逻辑动态 (控制) 系统中去. 例如, 文献 [100] 及 [38] 分别研究了一般有限值逻辑网络与切换布尔网络的稳定性与镇定. 文献 [101] 讨论了带脉冲的布尔网络的稳定性与镇定设计. 状态反馈镇定与输出反馈镇定分别在文献 [106] 与 [105] 中研究, 并且分别给出了反馈镇定器的设计方法.

第 14 章　布尔网络的辨识

14.1　网络的动态表达

一个标准的布尔网络可表示为

$$
\begin{cases}
x_1(t+1) = f_1(x_1(t), x_2(t), \cdots, x_n(t)), \\
x_2(t+1) = f_2(x_1(t), x_2(t), \cdots, x_n(t)), \\
\quad\vdots \\
x_n(t+1) = f_n(x_1(t), x_2(t), \cdots, x_n(t)).
\end{cases}
\tag{14.1.1}
$$

在向量形式下, 记

$$
x(t) = \ltimes_{i=1}^n x_i(t),
$$

设 $f_i, i = 1, \cdots, n$ 的结构矩阵为 $M_i \in \mathcal{L}_{2\times 2^n}$, $i = 1, \cdots, n$, 那么, (14.1.1) 可改写成

$$
\begin{cases}
x_1(t+1) = M_1 x(t), \\
x_2(t+1) = M_2 x(t), \\
\quad\vdots \\
x_n(t+1) = M_n x(t).
\end{cases}
\tag{14.1.2}
$$

(14.1.2) 称为系统 (14.1.1) 的分量代数形式. 将 (14.1.2) 所有方程相乘, 则可得到 (14.1.1) 的代数状态空间表示

$$
x(t+1) = Lx(t). \tag{14.1.3}
$$

从前面几章的讨论中可知 (14.1.1), (14.1.2), 和 (14.1.3) 只是三种等价的表示形式. 它们各有优点: (14.1.1) 表达了动态变量间的逻辑关系; (14.1.3) 是一个标准离散时间线性系统, 它可方便应用于对网络的分析与设计. 后面将看到, (14.1.2) 在辨识或重构系统时很方便.

下面讨论这几种表达形式的互换. 先考虑 (14.1.1) $\Rightarrow$ (14.1.2) $\Rightarrow$ (14.1.3):

先回忆两个小命题, 它们提供了必要的工具.

命题 14.1.1 1) 定义

$$
\begin{aligned}
D_f[i,j] &:= \mathbf{1}_i^{\mathrm{T}} \otimes I_j\\
D_r[i,j] &:= I_i \otimes \mathbf{1}_j^{\mathrm{T}}.
\end{aligned}
\tag{14.1.4}
$$

设 $X \in \Delta_i$, $Y \in \Delta_j$. 则

$$
\begin{aligned}
D_f[i,j]XY &= Y\\
D_r[i,j]XY &= X.
\end{aligned}
\tag{14.1.5}
$$

2) 定义

$$
D_m[i,j,k] := I_i \otimes \mathbf{1}_j^{\mathrm{T}} \otimes I_k. \tag{14.1.6}
$$

设 $X \in \Delta_i$, $Y \in \Delta_j$, $Z \in \Delta_k$. 则

$$
D_m[i,j,k]XYZ = XZ. \tag{14.1.7}
$$

注意到 $X \ltimes Y = X \otimes Y$, 上述命题是显见的.

命题 14.1.2 设

$$
y_j = M_j \ltimes_{i=1}^n x_i, \quad j = 1, \cdots, m. \tag{14.1.8}
$$

这里 $x_i \in \Delta_{k_i}$, $i = 1, \cdots, n$, $y_j \in \Delta_{r_j}$, $j = 1, \cdots, m$. 则有

$$
y = Mx, \tag{14.1.9}
$$

这里 $y = \ltimes_{j=1}^m y_j$, $x = \ltimes_{i=1}^n x_i$,

$$
M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{2^n \times 2^n}.
$$

其中, $*$ 是 Khatri-Rao 积.

下面结合一个例子来说明不同表示法的转换.

例 14.1.1 考虑布尔网络如图 14.1.1 所示.

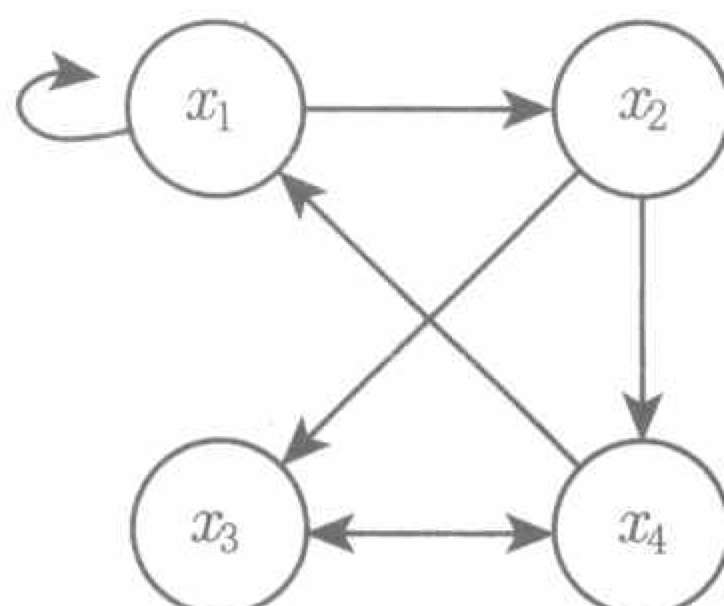


图 14.1.1 (14.1.10) 的网络图

网络的逻辑动态表达式为

$$
\begin{cases}
x_1(t+1) = x_1(t) \rightarrow x_4(t), \\
x_2(t+1) = \neg x_1(t), \\
x_3(t+1) = x_2(t) \wedge x_4(t), \\
x_4(t+1) = x_2(t) \leftrightarrow x_3(t).
\end{cases} \tag{14.1.10}
$$

我们先将 (14.1.10) 变为分量代数形式:

$$
\begin{cases}
x_1(t+1) = M_i x_1(t) x_4(t) = \delta_2[1\ 2\ 1\ 1] x_1(t) x_4(t), \\
x_2(t+1) = M_n x_1(t) = \delta_2[2\ 1] x_1(t), \\
x_3(t+1) = M_c x_2(t) x_4(t) = \delta_2[1\ 2\ 2\ 2] x_2(t) x_4(t), \\
x_4(t+1) = M_e x_2(t) x_3(t) = \delta_2[1\ 2\ 2\ 1] x_2(t) x_3(t).
\end{cases} \tag{14.1.11}
$$

这个分量代数形式与 (14.1.2) 不同, 因右边变量不全. 利用命题 14.1.1, 我们可以将其变量补全. 以第一个方程为例, 有

$$
x_1(t+1) = M_i D_m[2,4,2] x_1(t) x_2(t) x_3(t) x_4(t) := M_1 x(t),
$$

则

$$
\begin{aligned}
M_1 &= M_i D_m[2,4,2] \\
&= \delta_2[1,2,1,2,1,2,1,2,1,1,1,1,1,1,1,1].
\end{aligned}
$$

类似地, 可以得到

$$
\begin{aligned}
M_2 &= \delta_2[2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1]; \\
M_3 &= \delta_2[1,2,1,2,2,2,2,2,1,2,1,2,2,2,2,2]; \\
M_4 &= \delta_2[1,1,2,2,2,2,1,1,1,1,2,2,2,2,1,1].
\end{aligned}
$$

利用命题 14.1.2, 不难得到代数状态空间表示

$$
x(t+1) = L x(t), \tag{14.1.12}
$$

这里

$$
\begin{aligned}
L &= M_1 * M_2 * M_3 * M_4 \\
&= \delta_{16}[5,15,6,16,8,16,7,15,1,3,2,4,4,4,3,3].
\end{aligned}
$$

再考虑 (14.1.3) $\Rightarrow$ (14.1.2) $\Rightarrow$ (14.1.1):

假定 (14.1.3) 已知, 即状态转移阵 L 已知, 我们首先要将 (14.1.2) 中, 每个状态变量的动态系统 f_i 的结构矩阵 M_i 计算出来, 然后再将逻辑表达式 (14.1.1) 重构出来.

首先定义一组矩阵 $S_i^n \in \mathcal{L}_{2\times 2^n}$, $i=1,\cdots,n$, 称为重构矩阵, 如下:

$$
\begin{aligned}
S_1^n &= I_2 \otimes \mathbf{1}_{2^{n-1}}^{\mathrm{T}},\\
S_2^n &= \mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-2}}^{\mathrm{T}},\\
S_3^n &= \mathbf{1}_{2^2}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-3}}^{\mathrm{T}},\\
&\ \vdots\\
S_n^n &= \mathbf{1}_{2^{n-1}}^{\mathrm{T}} \otimes I_2.
\end{aligned}
\tag{14.1.13}
$$

重构矩阵在上一章讨论过, 它们的显式表示可见 (13.2.23).

下面的定理指出如何由 L 算出 $M_i, i=1,\cdots,n$.

定理 14.1.1 设 (14.1.1) 的代数状态空间表示为 (14.1.3). 那么, f_i 的结构矩阵 M_i 可由下式算出:

$$
M_i = S_i^n L, \quad i=1,2,\cdots,n. \tag{14.1.14}
$$

证明 回顾命题 14.1.1 不难发现

$$
S_i^n = D_m[2^{i-1}, 2, 2^{n-i}].
$$

因为

$$
x(t+1) = \ltimes_{j=1}^n x_j(t+1) = L \ltimes_{j=1}^n x_j(t). \tag{14.1.15}
$$

将 (14.1.15) 式左乘 S_i^n 即得

$$
x_i(t+1) = S_i^n L \ltimes_{j=1}^n x_j(t).
$$

由 M_i 的唯一性, 结论显见. □

注意到一个布尔网络的网络图是一个有向图. 对于它的一个结点 i, 指向它的邻边数称为 i 的入度(in-degree), 从它发出的邻边数称为出度(out-degree).

回到 (14.1.14), 那里给出的 M_i 是关于所有变量的乘积的系数阵. 这其实是不合理的, 因为很多变量可能与 f_i 无关. 一个大的网络, 其每个变量的演化, 其直接依赖的邻域变量, 也就是它的入度, 常常是很小的. 因此, 将 M_i 返回到逻辑表达式前, 最好将所有无关变量先删除.

命题 14.1.3 考察系统 (14.1.2). 如果 M_i 满足

$$
M_i W_{[2,2^{j-1}]}(M_n - I_2) = 0, \tag{14.1.16}
$$

那么, j 不属于 i 的邻域. 换言之, 不存在 $j \to i$ 的边. 而且, 关于 x_i 的方程可以用

$$
x_i(t+1) = M_i' x_1(t)\cdots x_{j-1}(t) x_{j+1}(t)\cdots x_n(t) \tag{14.1.17}
$$

来代替, 这里

$$M_i' = M_i W_{[2,2^{j-1}]} \delta_2^1.$$

证明 注意到可以将 (14.1.2) 的第 i 个方程改写成

$$x_i(t+1) = M_i W_{[2,2^{j-1}]} x_j(t) x_1(t) \cdots x_{j-1}(t) x_{j+1}(t) \cdots x_n(t).$$

现在将 $x_j(t)$ 用 $\neg x_j(t)$ 代替, 如果这不改变整个结构矩阵, 那就意味着 $x_i(t+1)$ 不依赖 $x_j(t)$. 这种替代的不变性显然可用 (14.1.16) 来表示. 至于 (14.1.17), 因为 $x_j(t)$ 不影响 $x_i(t+1)$, 可以直接令 $x_j(t) = \delta_2^1$ (当然, 如果你乐意, 你也可以用 $x_j(t) = \delta_2^2$ 来代替), 这样就得到上述表达式. □

注 重复使用 (14.1.16) 进行检验, 再用 (14.1.17) 化简, 就可以将所有的无关变量都从方程移走, 最后, 我们得到没有无关变量的逻辑动力系统, 称之为净形式.

下面给一个例子来说明.

例 14.1.2 假定有一个 5 结点的布尔网络

$$x(t+1) = Lx(t),$$

其网络矩阵 $L \in \mathcal{L}_{32\times 32}$ 为

$$\begin{aligned} L = \delta_{32}[&3\ 6\ 7\ 6\ 19\ 22\ 31\ 30\ 19\ 22\ 23\ 22\ 3\ 6\ 15\ 14\\ &3\ 5\ 7\ 5\ 19\ 21\ 31\ 29\ 19\ 21\ 23\ 21\ 3\ 5\ 15\ 13]. \end{aligned}$$

利用重构矩阵 S_i^5, 有

$$M_i = S_i^5 L, \quad i = 1, 2, 3, 4, 5,$$

它们是

$$\begin{aligned}
M_1 = \delta_2[&1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ \ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\\
&2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1];\\
M_2 = \delta_2[&1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\ \ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\\
&1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2];\\
M_3 = \delta_2[&1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ \ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\\
&1\ 2\ 2\ 2\ 1\ 2\ 2\ 2\ 1\ 2\ 2\ 2];\\
M_4 = \delta_2[&2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ \ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\\
&2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1];\\
M_5 = \delta_2[&1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ \ 1\ 2\ 1\ 2\ 1\ 1\ 1\ 1\\
&1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1].
\end{aligned}$$

先考察 M_1, 容易检验

$$\begin{cases} M_1 M_n = M_1, \\ M_1 W_{[2]} M_n \neq M_1, \\ M_1 W_{[2,2^2]} M_n \neq M_1, \\ M_1 W_{[2,2^3]} M_n = M_1, \\ M_1 W_{[2,2^4]} M_n = M_1. \end{cases}$$

我们得出结论: $x_1(t+1)$ 只依赖于 $x_2(t)$ 和 $x_3(t)$. 那么, 可以把无关变量 $x_1(t)$, $x_4(t)$, $x_5(t)$ 从第一个方程移走.

$$x_1(t+1) = M_1 x_1(t) x_2(t) x_3(t) x_4(t) x_5(t), \tag{14.1.18}$$

将 (14.1.8) 中的 $x_1(t)$, $x_4(t)$, $x_5(t)$ 用任何定常逻辑值代替. 例如, 令 $x_1(t) = x_4(t) = x_5(t) = \delta_2^1$. 那么可得

$$\begin{aligned} x_1(t+1) &= M_1 \delta_2^1 x_2(t) x_3(t) \delta_2^1 \delta_2^1 \\ &= M_1 W_{[4,8]} (\delta_2^1)^3 x_2(t) x_3(t) \\ &= \delta_2[1\ 2\ 2\ 1] x_2(t) x_3(t). \end{aligned} \tag{14.1.19}$$

类似地, 可以将其他方程中无关变量移去. 最后得到

$$\begin{cases} x_1(t+1) = \delta_2[1\ 2\ 2\ 1] x_2(t) x_3(t), \\ x_2(t+1) = \delta_2[1\ 1\ 1\ 2] x_3(t) x_4(t), \\ x_3(t+1) = \delta_2[1\ 2\ 2\ 2] x_4(t) x_5(t), \\ x_4(t+1) = \delta_2[2\ 1] x_5(t), \\ x_5(t+1) = \delta_2[1\ 2\ 1\ 1] x_1(t) x_5(t). \end{cases} \tag{14.1.20}$$

据此, 我们可以重构网络, 见图 14.1.2.

而且, 从网络的代数表达式, 不难得到逻辑方程如下:

$$\begin{cases} x_1(t+1) = x_2(t) \leftrightarrow x_3(t), \\ x_2(t+1) = x_3(t) \vee x_4(t), \\ x_3(t+1) = x_4(t) \wedge x_5(t), \\ x_4(t+1) = \neg x_5(t), \\ x_5(t+1) = x_1(t) \rightarrow x_5(t). \end{cases} \tag{14.1.21}$$

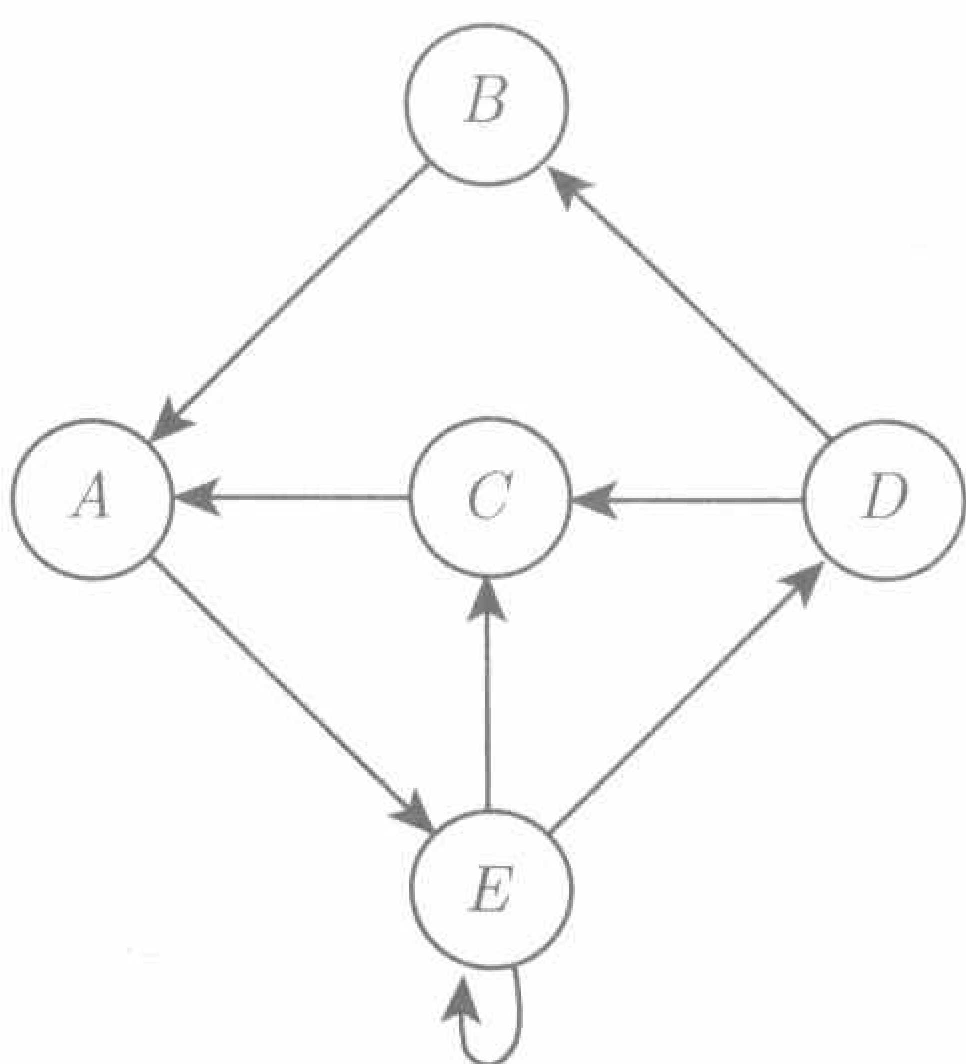


图 14.1.2　由网络矩阵重构网络图

注　与代数方程不同, 在一个逻辑表达式里直接判断那些变量是无关变量可能很困难. 于是, 我们先将逻辑形式转化为代数形式, 然后利用命题 14.1.3 来检验, 并进而将无关变量移走, 从而得到净形式, 最后将它返回 (较简单的) 逻辑形式. 当我们想从布尔网络的动态方程构造关联矩阵时, 动态方程必须保持净形式.

在例 14.1.2 中, 我们通过观查将系统从代数形式 (14.1.20) 转回逻辑形式 (14.1.11). 一般地说, 将逻辑方程从代数形式转回逻辑形式并非易事. 设

$$E = f(x_1, x_2, \cdots, x_n) = M_f x_1 x_2 \cdots x_n, \tag{14.1.22}$$

利用命题 12.3.1, 可以用一个递归算法得到它的逻辑表达式. 下面用一个例子说明这个算法.

例 14.1.3　设

$$E = \delta_2[1\ 2\ 2\ 1\ 2\ 1\ 2\ 1\ 1\ 1\ 2\ 2\ 2\ 1\ 1\ 2] x_1 x_2 x_3 x_4. \tag{14.1.23}$$

那么

$$E = [x_1 \wedge f_1(x_2, x_3, x_4)] \vee [\neg x_1 \wedge f_2(x_2, x_3, x_4)],$$

并且

$$M_{f_1} = \delta_2[1\ 2\ 2\ 1\ 2\ 1\ 2\ 1];$$
$$M_{f_2} = \delta_2[1\ 1\ 2\ 2\ 2\ 1\ 1\ 2].$$

下一步, 继续拆分:

$$f_1(x_2, x_3, x_4) = [x_2 \wedge f_{11}(x_3, x_4)] \vee [\neg x_2 \wedge f_{12}(x_3, x_4)],$$

这里

$$M_{f_{11}} = \delta_2[1\ 2\ 2\ 1] \quad \Rightarrow \quad f_{11}(x_3, x_4) = x_3 \leftrightarrow x_4;$$
$$M_{f_{12}} = \delta_2[2\ 1\ 2\ 1] \quad \Rightarrow \quad f_{12}(x_3, x_4) = \neg x_4.$$

$$f_2(x_2,x_3,x_4) = [x_2 \wedge f_{21}(x_3,x_4)] \vee [\neg x_2 \wedge f_{22}(x_3,x_4)],$$

这里

$$\begin{aligned} M_{f_{21}} &= \delta_2[1\ 1\ 2\ 2] \quad \Rightarrow \quad f_{21}(x_3,x_4) = x_3; \\ M_{f_{22}} &= \delta_2[2\ 1\ 1\ 2] \quad \Rightarrow \quad f_{22}(x_3,x_4) = \neg(x_3 \leftrightarrow x_4). \end{aligned}$$

把它们放到一起, 则得

$$\begin{aligned} E = {} & [x_1 \wedge x_2 \wedge (x_3 \leftrightarrow x_4)] \vee [x_1 \wedge (\neg x_2) \wedge (\neg x_4)] \vee [(\neg x_1) \wedge x_2 \wedge x_3] \\ & \vee [(\neg x_1) \wedge (\neg x_2) \wedge (\neg (x_3 \leftrightarrow x_4))]. \end{aligned}$$

注 考虑一个布尔控制网络, 其代数表达式为

$$\begin{cases} x(t+1) = Lu(t)x(t), & x(t) \in \Delta_{2^n},\ u(t) \in \Delta_{2^m}, \\ y(t) = Hx(t), & y(t) \in \Delta_{2^p}. \end{cases} \tag{14.1.24}$$

那么, 定理 14.1.1 和命题 12.3.1 可以用来将状态方程或输出方程转回逻辑表达形式.

14.2 一般网络的模型重构

设一布尔网络具有 n 个结点. 令 $X(t) = \{x_1(t), \cdots, x_n(t)\}$. 设观测到的数据为 $\{X(0), X(1), \cdots, X(N)\}$. 定义模型重构如下:

定义 14.2.1 设一组观测数据 $\{X(0), X(1), \cdots, X(N)\}$ 已知, 这里 $X(t) = \{x_1(t), \cdots, x_n(t)\}$. 模型重构问题指: 找到逻辑动态系统 (14.1.1), 使观测数据符合该系统.

一个符合观测数据的模型称为该数据的一个实现.

模型重构也称为辨识. 它曾被许多作者研究过, 例如, 文献 [107] 提出一种逆向工程算法来描述基因网络结构; 文献 [18] 提出一种用少量基因来辨识基因模式的方法; 文献 [19] 提出另一种基于矩阵乘法与指纹函数的辨识算法; 文献 [114] 提出一种要求较少平均时间的随机网络搜寻方法, 等.

根据定义不难得到下面的结果:

命题 14.2.1 系统是唯一可辨识的, 当且仅当, 数据 $\{X(0), X(1), \cdots, X(N-1)\}$ 包括所有的状态.

证明 利用 $x(t) := \ltimes_{i=1}^n x_i(t)$ 将数据转化为向量形式. 那么, 在代数形式下我们有 $x(t) = \delta_{2^n}^i$ 且 $x(t+1) = \delta_{2^n}^j$, 当且仅当 L 的第 i 列为

$$\mathrm{Col}_i(L) = \delta_{2^n}^j. \tag{14.2.1}$$

因此, L 是唯一可辨识的, 当且仅当在向量形式下

$$\{x(0), x(1), \cdots, x(N-1)\} = \Delta_{2^n}.$$

结论显见. □

如果观测数据有若干组, 则有如下结论.

推论 14.2.1 假定有 k 组观测数据

$$\{X^i(0), X^i(1), \cdots, X^i(N_i)\}, \quad i = 1, \cdots, k.$$

那么, 系统是唯一可辨识的, 当且仅当

$$\bigcup_{i=1}^{k} \{x^i(0), \cdots, x^i(N_i - 1)\} = \Delta_{2^n}. \tag{14.2.2}$$

注 1) 由命题 14.2.1 可知, 要辨识带 n 个结点的布尔网络至少需要 $2^n + 1$ 个数据点.

2) 当数据不满足命题 14.2.1 的要求时, 仍然可以利用 (14.2.1) 来辨识结构矩阵 L 的某些列. 这时模型不唯一确定. L 中未确定的列可任意取值.

例 14.2.1 设一布尔网络有 5 个结点. 12 组观测数据显示在图 14.2.1 中, 每行为一个独立观测数组, 其中标号为 1 的白圈表示健康细胞, 标号为 0 的黑圈表示被感染的细胞. 我们希望建立一个动态模型来刻画传播过程.

从第一组数据可得到 (这里编号先从左到右再从上到下)

$$\begin{array}{lll}
X^1(0) = (0,0,1,0,0); & X^1(1) = (0,1,1,1,0); & X^1(2) = (1,1,0,1,1);\\
X^1(3) = (0,1,1,0,0); & X^1(4) = (1,1,1,1,1); & X^1(5) = (1,1,1,0,0);\\
X^1(6) = (1,1,1,1,0); & X^1(7) = (1,1,0,1,0); & X^1(8) = (0,1,0,1,0);\\
X^1(9) = (0,1,0,1,1); & X^1(10) = (0,1,1,0,1); & X^1(11) = (1,1,0,0,1);\\
X^1(12) = (0,0,0,0,0); & X^1(13) = (0,0,1,1,0); & X^1(14) = (0,1,0,1,0).
\end{array}$$

现在, $X^1(0) = \delta_2\{2,2,1,2,2\}$ 于是

$$x^1(0) = \delta_2^2 \ltimes \delta_2^2 \ltimes \delta_2^1 \ltimes \delta_2^2 \ltimes \delta_2^2 = \delta_{32}^{28}.$$

类似地可算出

$$\begin{array}{lllll}
x^1(0) = \delta_{32}^{28}; & x^1(1) = \delta_{32}^{18}; & x^1(2) = \delta_{32}^{5}; & x^1(3) = \delta_{32}^{20}; & x^1(4) = \delta_{32}^{1};\\
x^1(5) = \delta_{32}^{4}; & x^1(6) = \delta_{32}^{2}; & x^1(7) = \delta_{32}^{6}; & x^1(8) = \delta_{32}^{22}; & x^1(9) = \delta_{32}^{21};\\
x^1(10) = \delta_{32}^{19}; & x^1(11) = \delta_{32}^{7}; & x^1(12) = \delta_{32}^{32}; & x^1(13) = \delta_{32}^{26}; & x^1(14) = \delta_{32}^{22}.
\end{array}$$

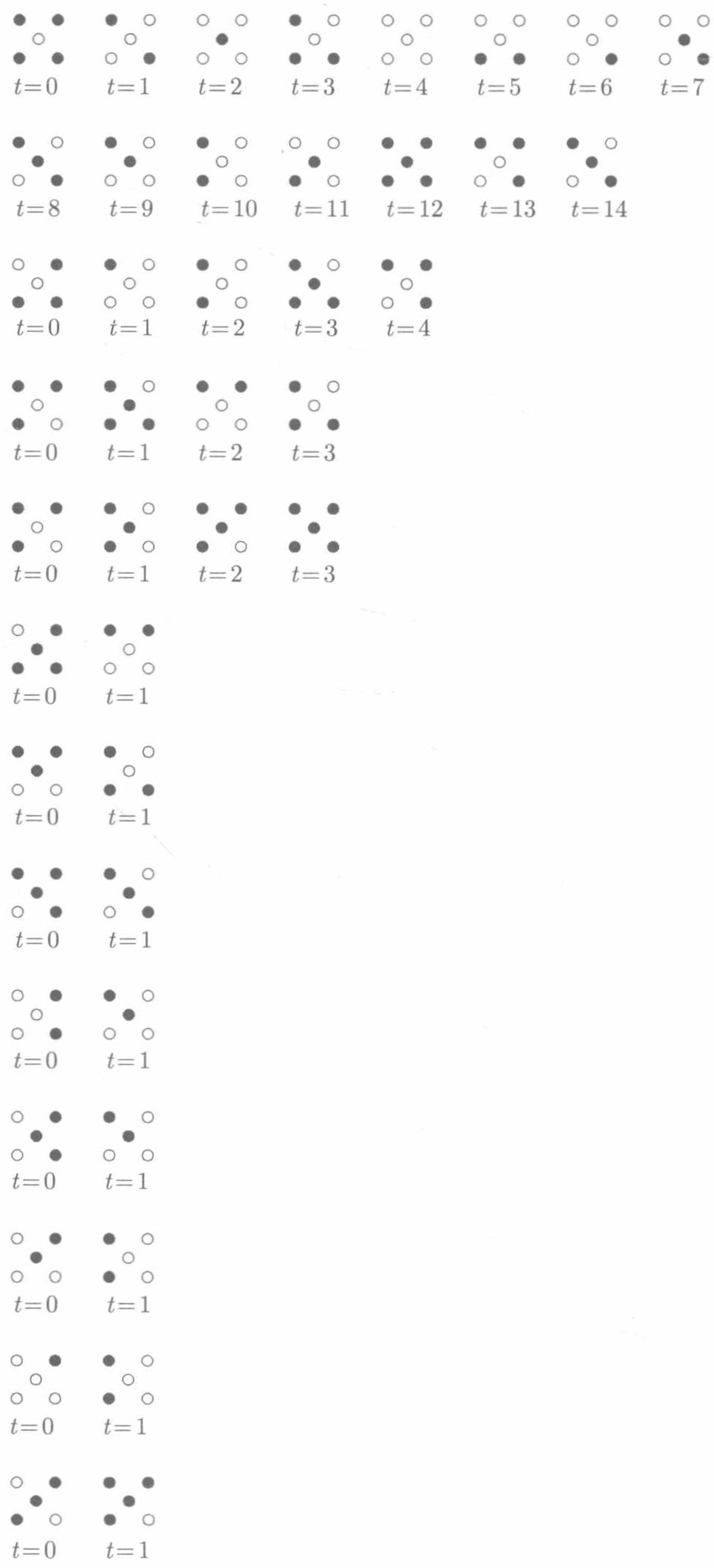


图 14.2.1　例 14.2.1 的观测数据

利用命题 14.2.1 (准确地说, 利用等式 (14.2.1)), 我们知道

$$\mathrm{Col}_{28}(L)=\delta_{32}^{18};\quad \mathrm{Col}_{18}(L)=\delta_{32}^{5};\quad \mathrm{Col}_{5}(L)=\delta_{32}^{20};\quad \cdots.$$

L 的 14 列被确定了.

对其他组数据如法泡制, 最后可得

$$\begin{aligned}L=\delta_{32}[&\,4\ 6\ 8\ 2\ 20\ 22\ 32\ 26\ 19\ 21\ 23\ 17\ 19\ 21\ 31\ 25\\ &3\ 5\ 7\ 1\ 19\ 21\ 31\ 25\ 20\ 22\ 24\ 18\ 20\ 22\ 32\ 26].\end{aligned}\tag{14.2.3}$$

因此可由观测数据得到传播过程的动态方程为

$$x(t+1)=Lx(t),\tag{14.2.4}$$

这里 $L\in\mathcal{L}_{32\times 32}$ 可见 (14.2.3).

下面构造其逻辑动态方程以观测其结点的相互作用. 利用 (14.1.4), 相应的重构算子为

$$\begin{aligned}
S_1^5&=\delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2],\\
S_2^5&=\delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2],\\
S_3^5&=\delta_2[1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2],\\
S_4^5&=\delta_2[1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2],\\
S_5^5&=\delta_2[1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2].
\end{aligned}\tag{14.2.5}$$

于是我们可以得到分量代数表达式如下:

$$\begin{cases}x_1(t)=M_1x(t),\\ x_2(t)=M_2x(t),\\ x_3(t)=M_3x(t),\\ x_4(t)=M_4x(t),\\ x_5(t)=M_5x(t),\end{cases}\tag{14.2.6}$$

这里

$$\begin{aligned}
M_1&=S_1^5L=\delta_2[1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2],\\
M_2&=S_2^5L=\delta_2[1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2],\\
M_3&=S_3^5L=\delta_2[1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1\ 1\ 2\ 2\ 1],\\
M_4&=S_4^5L=\delta_2[2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2\ 1],\\
M_5&=S_5^5L=\delta_2[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2].
\end{aligned}\tag{14.2.7}$$

先考虑 $x_1(t)$.

$x_1(t+1) = M_1x(t) = \delta_2[1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2]x(t)$.
利用命题 14.1.2 可移去无关变量. 容易检验

$$
\begin{aligned}
M_1(M_n - I_2) &= 0,\\
M_1W_{[2,2]}(M_n - I_2) &\neq 0,\\
M_1W_{[2,4]}(M_n - I_2) &\neq 0,\\
M_1W_{[2,8]}(M_n - I_2) &= 0,\\
M_1W_{[2,16]}(M_n - I_2) &= 0.
\end{aligned}
\tag{14.2.8}
$$

因此, $x_1(t), x_4(t), x_5(t)$ 是 $x_1(t+1)$ 动态表达式中的无关变量. 令 $x_1(t) = x_4(t) = x_5(t) = \delta_2^1$, 则得

$$
\begin{aligned}
x_1(t+1) &= M_1x_1(t)x_2(t)x_3(t)x_4(t)x_5(t)\\
&= M_1W_{[4,8]}x_4(t)x_5(t)x_1(t)x_2(t)x_3(t)\\
&= M_1W_{[4,8]}(\delta_2^1)^3x_2(t)x_3(t)\\
&= \delta_2[1\ 2\ 2\ 2]x_2(t)x_3(t).
\end{aligned}
$$

因此, 其逻辑表达式为

$$
x_1(t+1) = x_2(t) \wedge x_3(t).
$$

同样处理 $x_2(t), x_3(t), x_4(t)$ 以及 $x_5(t)$. 最后, 得到整个网络的逻辑表达式

$$
\begin{cases}
x_1(t+1) = x_2(t) \wedge x_3(t),\\
x_2(t+1) = x_3(t) \vee x_4(t),\\
x_3(t+1) = x_4(t) \leftrightarrow x_5(t),\\
x_4(t+1) = \neg x_5(t),\\
x_5(t+1) = x_1(t) \bar{\vee} x_2(t).
\end{cases}
\tag{14.2.9}
$$

图 14.2.2 为此系统的网络图.

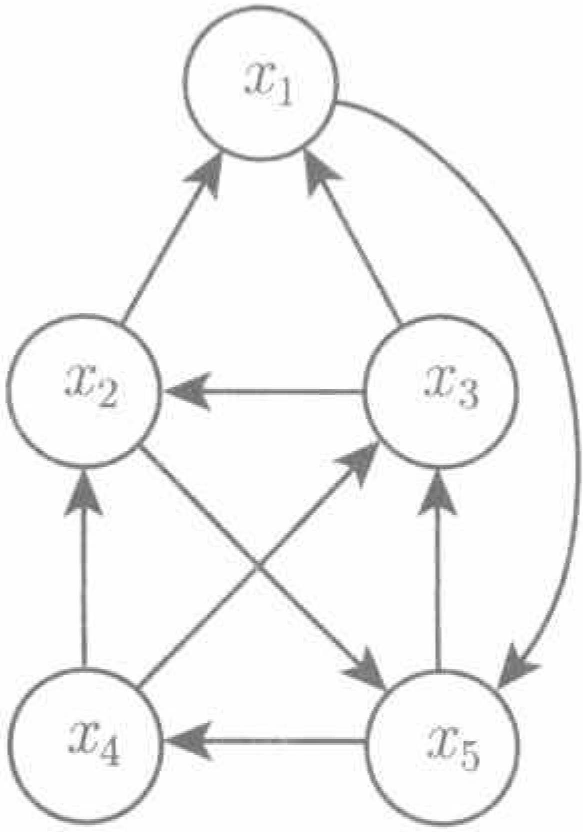


图 14.2.2　系统 (14.2.9) 的网络图

14.3　基于网络图的重构

上节讨论了从观测数据到系统的重构的一般方法. 同时指出, 要唯一确定系统模型至少需要 2^n+1 个数据点. 如果 n 不是个小数, 那么需要的数据是很大的. 幸运的是, 在许多实际场合, 系统的网络图是已知的. 这时, 需要的数据可大大减少.

注意到当网络较大时画出它的网络图是困难的. 另一种表示结点间动力学关联性的方法是给出关联矩阵[125]. 回忆关联矩阵的定义: 设网络有 n 个结点, $\mathcal{J}=(r_{i,j})\in\mathcal{M}_{n\times n}$ 称为其关联矩阵, 这里, $r_{i,j}=1$, 表明 $x_i(t+1)$ 直接依赖于 $x_j(t)$; 否则, $r_{i,j}=0$.

考察下面的例子.

例 14.3.1　考虑一个 8 个结点的网络, 其网络图见图 14.3.1.

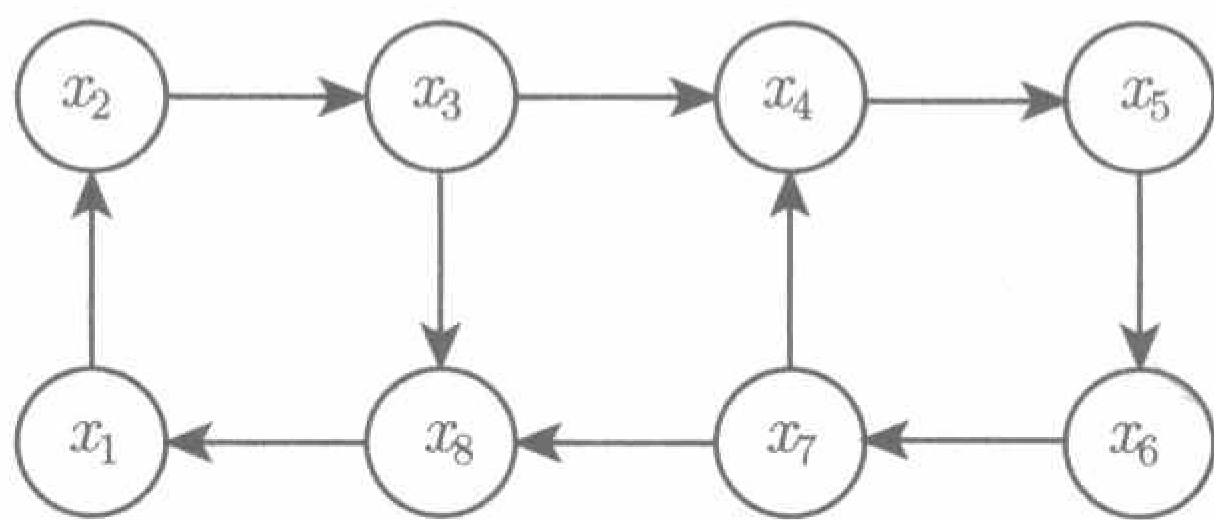


图 14.3.1　系统 (14.3.1) 的网络图

设计这个网络我们有图 14.3.2 所示观测数据.

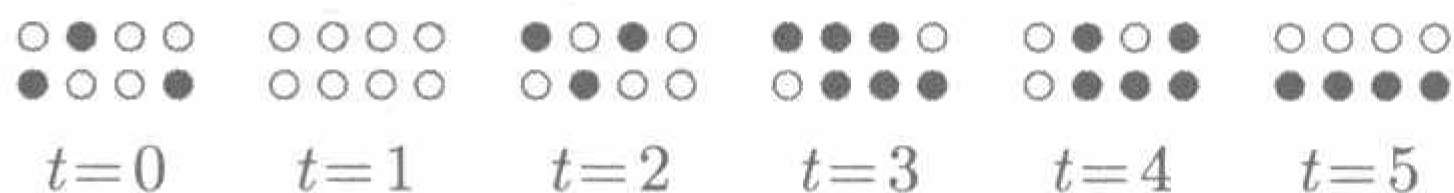


图 14.3.2　系统 14.3.1 的观测数据

我们用分量代数模式辨识其动力学模型. 即令

$$\begin{cases} x_1(t+1)=M_1x_8(t),\\ x_2(t+1)=M_2x_1(t),\\ x_3(t+1)=M_3x_2(t),\\ x_4(t+1)=M_4x_3(t)x_7(t),\\ x_5(t+1)=M_5x_4(t),\\ x_6(t+1)=M_6x_5(t),\\ x_7(t+1)=M_7x_6(t),\\ x_8(t+1)=M_8x_3(t)x_7(t). \end{cases} \tag{14.3.1}$$

根据观测数据易知

$$x_8(0)=0 \Rightarrow x_1(1)=1;$$
$$x_8(1)=1 \Rightarrow x_1(2)=0; \cdots .$$

在向量形式下则有

$$\mathrm{Col}_2(M_1)=\delta_2^1; \quad \mathrm{Col}_1(M_1)=\delta_2^2; \quad \cdots .$$

显见 $M_1=\delta_2[2\ 1]$, 因此可得

$$x_1(t+1)=\neg x_8(t).$$

类此, 可得到其他 M_i, $i=2,3,\cdots,8$. 最后, 可找出动力学方程如下:

$$\begin{cases} x_1(t+1)=\neg x_8(t), \\ x_2(t+1)=x_1(t), \\ x_3(t+1)=\neg x_2(t), \\ x_4(t+1)=x_3(t)\vee x_7(t), \\ x_5(t+1)=x_4(t), \\ x_6(t+1)=\neg x_5(t), \\ x_7(t+1)=x_6(t), \\ x_8(t+1)=x_3(t)\wedge x_7(t). \end{cases} \tag{14.3.2}$$

比较例 14.3.1 与例 14.2.1, 显然当网络图已知时 (等价地, 当关联矩阵已知时), 辨识所需数据可以大大减少.

14.4 最小入度建模

考察一个 n 结点布尔网络. 记第 k 个结点的入度为 $d_i(k)$, 它表明直接指向结点 k 的边数为 $d_i(k)$. 回忆邻接矩阵, 可知

$$d_i(k)=\sum_{j=1}^{n} r_{kj}, \quad k=1,\cdots,n. \tag{14.4.1}$$

实验表明, 在一个有序的网络中, 入度远小于结点数. 例如在一个 $n=100000$ 的随机灯泡网络里, 入度 $d_i=2$[89]. 本节考虑从观测数据得到一个入度最小的模型.

定义 14.4.1　考虑一个 n 结点网络. 一个入度为 $d_i^*(k)$, $k=1,\cdots,n$ 的实现称为最小入度实现, 如果对任何一个实现, 其入度为 $d_i(k)$, $k=1,\cdots,n$, 均有

$$d_i^*(k) \leqslant d_i(k), \quad k=1,\cdots,n.$$

显然, 最小入度实现需要的数据少得多. 而且, 前面的讨论说明, 这种实现有很强的合理性. 下面考虑如何得到最小入度实现. 记实验数据为 $\{X(0), X(1), \cdots, X(N)\}$. 考虑第 i 个结点, 设其方程为

$$x_i(t+1) = M_i x(t), \quad \text{这里} \quad M_i \in \mathcal{L}_{2\times 2^n}. \tag{14.4.2}$$

根据观测数据, M_i 的某些列可被确定. 不妨设

$$M_i = [* \ \cdots \ * \ c_{i_1} \ * \ \cdots \ * \ c_{i_2} \ * \ \cdots \ * \ \cdots \ * \ c_{i_s} \ * \ \cdots \ *], \tag{14.4.3}$$

这里 c_{i_j}, $j=1,\cdots,s$ 为已被辨识的列, $*$ 表示无法确定的列. 称 (14.4.3) 不确定结构矩阵. 下面定义一组矩阵

$$M_{i,j} := M_i W_{[2,2^{j-1}]}, \quad j=1,2,\cdots,n.$$

然后将得到的每个矩阵分为左右相等的两个子块如下:

$$M_{i,j} = \begin{bmatrix} M_{i,j}^1 & M_{i,j}^2 \end{bmatrix}. \tag{14.4.4}$$

那么, 有如下结论:

命题 14.4.1　f_i 具有一个不依赖于 x_j 的实现, 当且仅当

$$M_{i,j}^1 = M_{i,j}^2 \tag{14.4.5}$$

对不确定列有解.

证明　当 $j=1$, 有 $M_{i,j} = M_i$. 于是

$$M_i = \begin{bmatrix} M_i^1 & M_i^2 \end{bmatrix}. \tag{14.4.6}$$

如果

$$M_i^1 = M_i^2 \tag{14.4.7}$$

对不确定元素有解, 则能满足 $M_i^1 = M_i^2$. 根据等式 (14.1.6), 显见这个实现与 x_1 无关. 考虑 x_j, 将等式 (14.4.2) 改写成

$$x_i(t+1) = M_{i,j} x_j \ltimes_{k=1}^{j-1} x_k \ltimes_{k=j+1}^{n} x_k.$$

类似对 x_1 的讨论即得结论. □

下面给出一个最小入度实现的算法.

算法 14.4.1 第 1 步: 对每个分量代数方程, 用观测数据确定其某些列, 得到 (14.4.3). 定义指标集 $S_i = \{1, 2, \cdots, n\}$, $i = 1, \cdots, n$.

第 2 步: 构造 (14.4.6) 以检验 (14.4.7) 是否有解. 如果“有”, 用解代替待定元, 得到

$$x_i(t+1) = M_i^1 \ltimes_{j=2}^n x_j(t).$$

进行下一步.

第 j 步: (对 $3 \leqslant j \leqslant n$ 重复这一步) 检验 (14.4.5) 是否有解. 如果“有”, 用解代替待定元, 得到

$$x_i(t+1) = M_{i,j}^1 \ltimes_{1 \leqslant k \leqslant j-1,\, k \in S_i} x_k \ltimes_{k=j+1}^n x_k. \tag{14.4.8}$$

将 S_i 换为 $S_i \backslash \{j\}$.

下面的结论来自算法的构造.

命题 14.4.2 算法 14.4.1 产生最小入度实现.

我们用一个例子来描述这个算法.

例 14.4.1 考虑一组观测数据如图 14.4.1.



图 14.4.1 例 14.4.1 的观测数据

数据的向量形式为

$$x(0) = \delta_{16}^{12}; \quad x(1) = \delta_{16}^{16}; \quad x(2) = \delta_{16}^{8};$$
$$x(3) = \delta_{16}^{2}; \quad x(4) = \delta_{16}^{10}; \quad x(5) = \delta_{16}^{12}.$$

利用上节方法, 可辨识部分列如下:

$$M_1 = \delta_2[* \ 2 \ * \ * \ * \ * \ * \ 1 \ * \ 2 \ * \ 2 \ * \ * \ * \ 1].$$

设 $M_1^1 = M_1^2$, 可得到部分解如下:

$$M_1^1 = M_1^2 = \delta_2[* \ 2 \ * \ 2 \ * \ * \ * \ 1].$$

于是系统可简化为

$$x_1(t+1) = \delta_2[* \ 2 \ * \ 2 \ * \ * \ * \ 1] x_2(t) x_3(t) x_4(t).$$

将 M_1^1 分为两部分可得以下方程

$$\delta_2[* \ 2 \ * \ 2] = \delta_2[* \ * \ * \ 1],$$

这方程无解. 即, 方程依赖于 x_2.

考虑

$$M_{1,2} = M_1^1 W_{[2,2]} = \delta_2[* \ 2 \ * \ * \ * \ 2 \ * \ 1].$$

那么,

$$\delta_2[* \ 2 \ * \ *] = \delta_2[\ * \ 2 \ * \ 1]$$

有解

$$\delta_2[* \ 2 \ * \ 1].$$

原方程可转化为

$$x_1(t+1) = \delta_2[* \ 2 \ * \ 1]x_2(t)x_4(t).$$

最后考察 $x_4(t)$. 由于

$$\delta_2[* \ 2 \ * \ 1]W_{[2,2]} = \delta_2[* \ * \ 2 \ 1],$$

而

$$\delta_2[* \, *] = \delta_2[2 \ 1]$$

有解

$$\delta_2[2 \ 1].$$

最后, 有

$$x_1(t+1) = \delta_2[2 \ 1]x_2(t).$$

即

$$x_1(t+1) = \neg x_2(t).$$

将这套方法用于其他几个方程, 最后可得到, 系统的动力学方程为

$$\begin{cases} x_1(t+1) = \neg x_2(t), \\ x_2(t+1) = x_4(t) \vee x_1(t), \\ x_3(t+1) = x_1(t), \\ x_4(t+1) = x_3(t) \bar{\vee} x_4(t). \end{cases} \tag{14.4.9}$$

该系统的网络图见图 14.4.2.

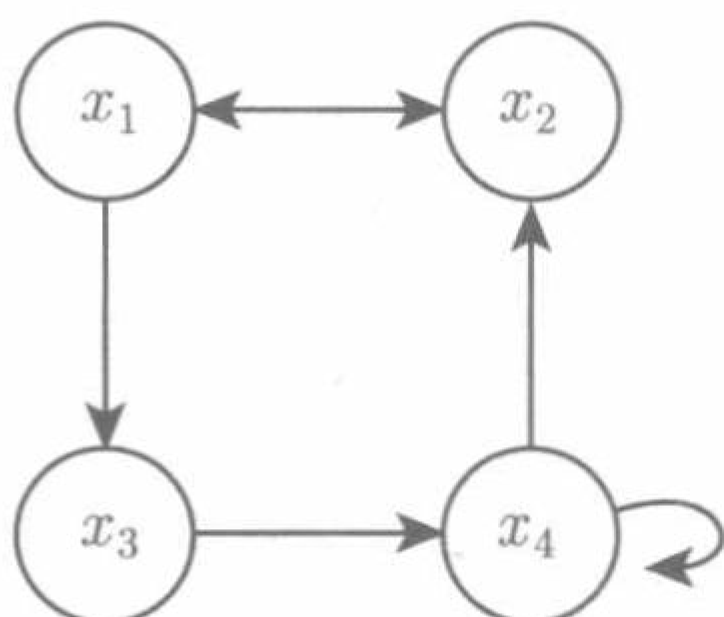


图 14.4.2　系统 (14.4.9) 的网络图

事实上, 如果

$$d_i(k) \leqslant \mu, \quad k = 1, \cdots, n,$$

那么, 容易看出, 辨识系统所需最小数据量为 $2^\mu + 1$, 一般情况下这比 $2^n + 1$ 少多了.

14.5 一致布尔网络的辨识

本节考虑具有一致动态结构的布尔网络的辨识问题. 它的物理意义如下: 假定有一组细胞. 每个细胞都可能受其邻居感染. 而且, 被感染的规律都一样. 那么, 每个结点与它的邻居之间的动态演化规律是一样的, 我们将这样的布尔网络称为一致布尔网络.

例 14.5.1 设一个网络的观测数据如图 14.5.1. 细胞 x_0 受它的邻居细胞 x_1, x_2, x_3, x_4, x_5, 和 x_6 感染, 其演化方程为

$$x_0(t+1) = f(x_1(t), x_2(t), x_3(t), x_4(t), x_5(t), x_6(t)). \tag{14.5.1}$$

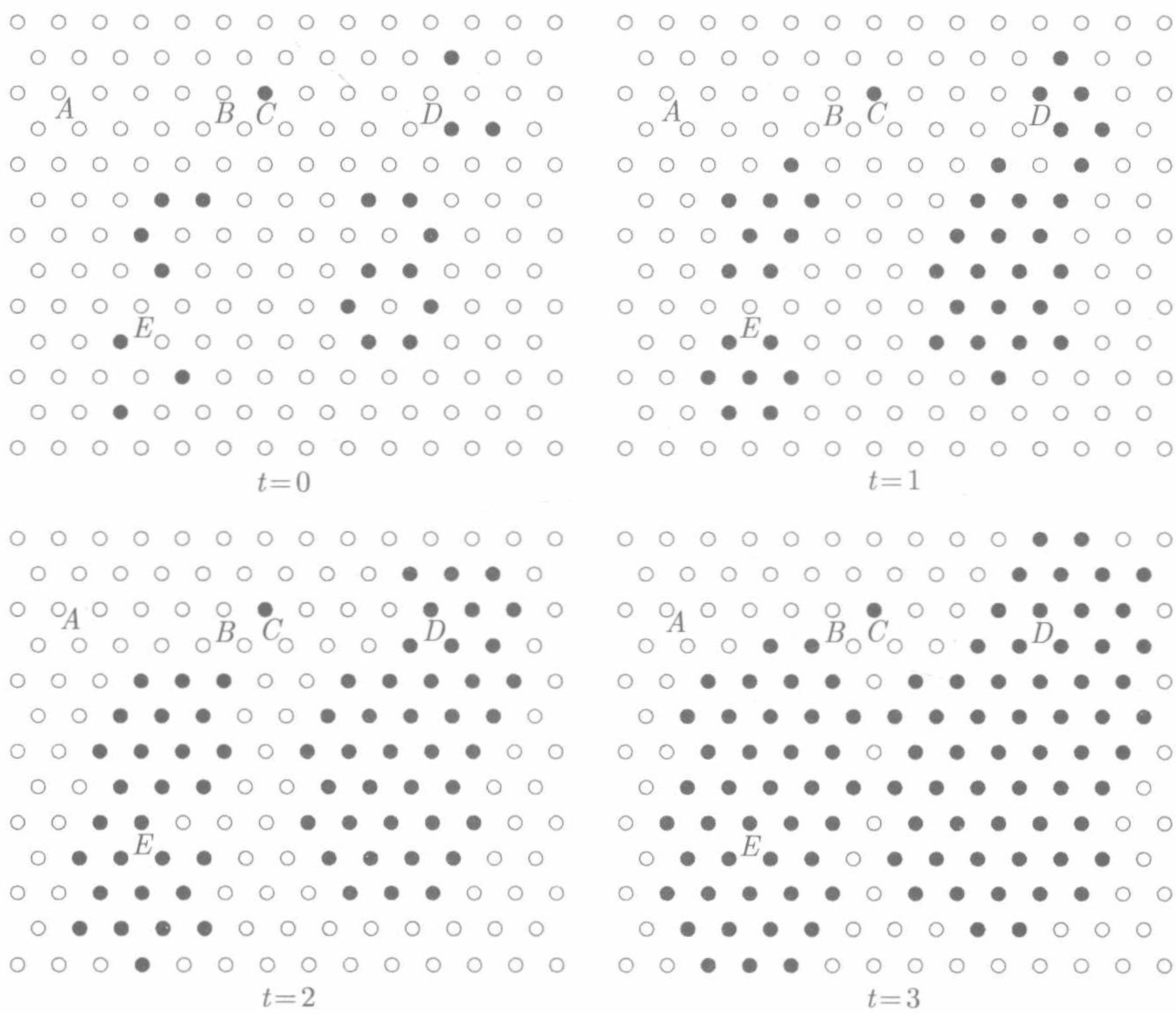


图 14.5.1 例 14.5.1 的观测数据

假定每个细胞 x_0 受它的邻居细胞感染的模式是一样的. 而且, 假定感染是各向同性的. 即, 设 $\sigma = (1,2,3,4,5,6) \in | S_6$, $G = \langle\sigma\rangle$ 是由 σ 生成的循环群, 则

$$f(x_1,x_2,x_3,x_4,x_5,x_6) = f(x_{\mu(1)},x_{\mu(2)},x_{\mu(3)},x_{\mu(4)},x_{\mu(5)},x_{\mu(6)}), \quad \forall \mu \in G. \quad (14.5.2)$$

将动态方程 (14.5.1) 表示为其代数形式

$$x_0(t+1) = M \ltimes_{i=0}^{6} x_i(t), \quad 这里 \quad M \in \mathcal{L}_{2\times 2^7}. \quad (14.5.3)$$

我们的目的是确定 M.

我们先用某些特殊位置来确定 M 的某些列.

1) 设 $x_0 = A$. 注意到 $x_0(0) = 1$ 而对其邻域有 $x_1(0) = 1$, $x_2(0) = 1$, $x_3(0) = 1$, $x_4(0) = 1$, $x_5(0) = 1$, 及 $x_6(0) = 1$, 于是, $X(0) = (1\ 1\ 1\ 1\ 1\ 1\ 1) \sim \delta_{128}^1$. 由于 $x_0(1) = 1$, 可得 $\mathrm{Col}_1(M) = \delta_2^1$;

2) 设 $x_0 = B$. 有 $x_0(0) = 1$ 且其邻域状态为 $x_1(0) = 1$, $x_2(0) = 1$, $x_3(0) = 1$, $x_4(0) = 1$, $x_5(0) = 1$, 及 $x_6(0) = 0$, 于是, $X(0) = (1\ 1\ 1\ 1\ 1\ 1\ 0) \sim \delta_{128}^2$. 由于 $x_0(1) = 1$, 可得 $\mathrm{Col}_2(M) = \delta_2^1$. 而且, 由各向同性可假定 $X(0)$ 处于各个旋转位置, 分别确定 $x_0(1)$. 即

$$\begin{aligned}
X(0) &= (1\,1\,1\,1\,1\,0\,1) \sim \delta_{128}^3 \quad \Rightarrow \quad \mathrm{Col}_3(M) = \delta_2^1,\\
X(0) &= (1\,1\,1\,1\,0\,1\,1) \sim \delta_{128}^5 \quad \Rightarrow \quad \mathrm{Col}_5(M) = \delta_2^1,\\
X(0) &= (1\,1\,1\,0\,1\,1\,1) \sim \delta_{128}^9 \quad \Rightarrow \quad \mathrm{Col}_9(M) = \delta_2^1,\\
X(0) &= (1\,1\,0\,1\,1\,1\,1) \sim \delta_{128}^{17} \quad \Rightarrow \quad \mathrm{Col}_{17}(M) = \delta_2^1,\\
X(0) &= (1\,0\,1\,1\,1\,1\,1) \sim \delta_{128}^{33} \quad \Rightarrow \quad \mathrm{Col}_{33}(M) = \delta_2^1.
\end{aligned}$$

3) 设 $x_0 = C$. 由于 $x_0(0) = 0$ 及 $x_0(1) = 0$, 有 $X(0) = (0,1,1,1,1,1,1) \sim \delta_{128}^{65}$, 即可得 $\mathrm{Col}_{65}(M) = \delta_2^2$.

4) 考虑 $x_0 = D$. 由于 $x_0(0) = 1$ 及 $x_0(1) = 0$, 利用各向同性可知

$$\begin{aligned}
X(0) &= (1\,0\,1\,0\,1\,1\,1) \sim \delta_{128}^{41} \quad \Rightarrow \quad \mathrm{Col}_{41}(M) = \delta_2^2,\\
X(0) &= (1\,1\,0\,1\,1\,1\,0) \sim \delta_{128}^{18} \quad \Rightarrow \quad \mathrm{Col}_{18}(M) = \delta_2^2,\\
X(0) &= (1\,0\,1\,1\,1\,0\,1) \sim \delta_{128}^{35} \quad \Rightarrow \quad \mathrm{Col}_{35}(M) = \delta_2^2,\\
X(0) &= (1\,1\,1\,1\,0\,1\,0) \sim \delta_{128}^{6} \quad \Rightarrow \quad \mathrm{Col}_{6}(M) = \delta_2^2,\\
X(0) &= (1\,1\,1\,0\,1\,0\,1) \sim \delta_{128}^{11} \quad \Rightarrow \quad \mathrm{Col}_{11}(M) = \delta_2^2,\\
X(0) &= (1\,1\,0\,1\,0\,1\,1) \sim \delta_{128}^{21} \quad \Rightarrow \quad \mathrm{Col}_{21}(M) = \delta_2^2.
\end{aligned}$$

5) 考虑 $x_0 = E$. 由于 $x_0(0) = 1$, $x_0(1) = 1$, 且 $x_0(2) = 0$, 有

$$\begin{aligned}
X(0) &= (1\,0\,1\,1\,0\,1\,1) \sim \delta_{128}^{37} \quad \Rightarrow \quad \mathrm{Col}_{37}(M) = \delta_2^1,\\
X(0) &= (1\,1\,1\,0\,1\,1\,0) \sim \delta_{128}^{10} \quad \Rightarrow \quad \mathrm{Col}_{10}(M) = \delta_2^1,\\
X(0) &= (1\,1\,0\,1\,1\,0\,1) \sim \delta_{128}^{19} \quad \Rightarrow \quad \mathrm{Col}_{19}(M) = \delta_2^1;
\end{aligned}$$

以及

$$
\begin{aligned}
X(0) = (1\,0\,1\,0\,0\,1\,1) \sim \delta_{128}^{45} \quad &\Rightarrow \quad \mathrm{Col}_{45}(M) = \delta_2^2,\\
X(0) = (1\,1\,0\,0\,1\,1\,0) \sim \delta_{128}^{26} \quad &\Rightarrow \quad \mathrm{Col}_{26}(M) = \delta_2^2,\\
X(0) = (1\,0\,0\,1\,0\,0\,1) \sim \delta_{128}^{51} \quad &\Rightarrow \quad \mathrm{Col}_{51}(M) = \delta_2^2,\\
X(0) = (1\,0\,1\,1\,0\,1\,0) \sim \delta_{128}^{38} \quad &\Rightarrow \quad \mathrm{Col}_{38}(M) = \delta_2^2,\\
X(0) = (1\,1\,1\,0\,1\,0\,0) \sim \delta_{128}^{12} \quad &\Rightarrow \quad \mathrm{Col}_{12}(M) = \delta_2^2,\\
X(0) = (1\,1\,0\,1\,0\,0\,1) \sim \delta_{128}^{23} \quad &\Rightarrow \quad \mathrm{Col}_{23}(M) = \delta_2^2.
\end{aligned}
$$

6) 继续这个过程, 最后 M 所有的列均可辨识出来:

$$
\begin{aligned}
M = \delta_2[&1\,1\,1\,2\,1\,2\,2\,2\,1\,1\,2\,2\,2\,2\,2\,2\,1\,2\,1\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\\
&1\,2\,2\,2\,1\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\\
&2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\\
&2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2\,2].
\end{aligned}
$$

略去详细推演, 最后可以从代数形式返回逻辑形式为

$$
\begin{aligned}
x_0(t+1) = x_0(t) &\wedge (x_1(t) \vee x_2(t)) \wedge (x_2(t) \vee x_3(t)) \wedge (x_3(t) \vee x_4(t))\\
&\wedge (x_4(t) \vee x_5(t)) \wedge (x_5(t) \vee x_6(t)) \wedge (x_6(t) \vee x_1(t))\\
&\wedge (x_1(t) \vee x_3(t)) \wedge (x_3(t) \vee x_5(t)) \wedge (x_5(t) \vee x_1(t))\\
&\wedge (x_2(t) \vee x_4(t)) \wedge (x_4(t) \vee x_6(t)) \wedge (x_6(t) \vee x_2(t)).
\end{aligned} \tag{14.5.4}
$$

14.6 带错误数据的辨识

在前面讨论的辨识问题中, 总假设数据完全准确. 虽然布尔变量不会有实变数那样的小误差, 但在实际问题中还可能出现错误数据. 本节讨论带错误数据的模型辨识.

由量测等可能引起的错误, 数据可能互相矛盾. 矛盾数据可能引起对立的辨认结果:

$$
\mathrm{Col}_i(L) = \begin{cases} \delta_{2^n}^p, & k \text{ 次},\\ \delta_{2^n}^q, & s \text{ 次}. \end{cases} \tag{14.6.1}
$$

1) 如果 $k \gg s$, 那么, 可以忽略 $\delta_{2^n}^q$ 而令 $\mathrm{Col}_i(L) = \delta_{2^n}^p$.

2) 如果 $k \ll s$, 那么, 可以忽略 $\delta_{2^n}^p$ 而令 $\mathrm{Col}_i(L) = \delta_{2^n}^q$.

3) 如果 $k \approx s$, 那么, 可能需要更多数据; 或者说, 这个确定性模型不可接受.

考虑每一个变量 x_i 的方程, 对于其结构矩阵 M_i 可作类似判定. 例如, 考虑最小入度辨识. 设 $\{I,J\}$ 为 $\{1,2,\cdots,2^n\}$ 的一个分割. 如果

$$\begin{cases}\mathrm{Col}_j(M)=\delta_2^{s_j}, & k_j \geqslant k,\ j\in J,\\ \mathrm{Col}_i(M)=\delta_2^{s_i}, & k_i \ll k,\ i\in I.\end{cases} \tag{14.6.2}$$

那么可以认为 $\mathrm{Col}_i(M)$, $i\in I$ 是一个错误列, 而设 $\mathrm{Col}_i(M)=*$, 即, 认为它们是不确定列.

换言之, 只取可信的数据来做辨识. 许多统计学的分法可能用来判断数据是否可靠. 下面给一个简单例子.

例 14.6.1　设有一个 3 结点布尔网络. 观测数据在图 14.6.1 中给出.

○●○ $t=0$　○○● $t=1$　●●○ $t=2$　●○● $t=3$　●○○ $t=4$　○○○ $t=5$　○●○ $t=6$　●●○ $t=7$　●○● $t=8$　●○○ $t=9$

○○○ $t=10$　●●● $t=11$　●○● $t=12$　●○○ $t=13$　○○○ $t=14$　○●○ $t=15$　●●○ $t=16$　●○● $t=17$　●○○ $t=18$　○○○ $t=19$

○●● $t=20$　●●○ $t=21$　●○● $t=22$　●○○ $t=23$　○○○ $t=24$　○●○ $t=25$　●●○ $t=26$　●○● $t=27$　●○○ $t=28$　○○○ $t=29$

●●● $t=30$　●○● $t=31$　●○○ $t=32$　○○○ $t=33$　○○● $t=34$　●●○ $t=35$　●○● $t=36$　●○○ $t=37$　○○○ $t=38$　○●○ $t=39$

●●○ $t=40$　●○● $t=41$　●○● $t=42$　○○○ $t=43$　○●○ $t=44$　○●● $t=45$　●●○ $t=46$　●○● $t=47$　●○○ $t=48$　○○○ $t=49$

图 14.6.1　例 14.6.1 的观测数据

这里, 50 个观测数据可表为向量形式如下:

$$\begin{array}{lllll}
x(0)=\delta_8^3; & x(1)=\delta_8^2; & x(2)=\delta_8^7; & x(3)=\delta_8^6; & x(4)=\delta_8^5;\\
x(5)=\delta_8^1; & x(6)=\delta_8^3; & x(7)=\delta_8^7; & x(8)=\delta_8^6; & x(9)=\delta_8^5;\\
x(10)=\delta_8^1; & x(11)=\delta_8^8; & x(12)=\delta_8^6; & x(13)=\delta_8^5; & x(14)=\delta_8^1;\\
x(15)=\delta_8^3; & x(16)=\delta_8^7; & x(17)=\delta_8^6; & x(18)=\delta_8^5; & x(19)=\delta_8^1;\\
x(20)=\delta_8^4; & x(21)=\delta_8^7; & x(22)=\delta_8^6; & x(23)=\delta_8^5; & x(24)=\delta_8^1;\\
x(25)=\delta_8^3; & x(26)=\delta_8^7; & x(27)=\delta_8^6; & x(38)=\delta_8^5; & x(29)=\delta_8^1;\\
x(30)=\delta_8^8; & x(31)=\delta_8^6; & x(32)=\delta_8^5; & x(33)=\delta_8^1; & x(34)=\delta_8^2;\\
x(35)=\delta_8^7; & x(36)=\delta_8^6; & x(37)=\delta_8^5; & x(38)=\delta_8^1; & x(39)=\delta_8^3;\\
x(40)=\delta_8^7; & x(41)=\delta_8^6; & x(42)=\delta_8^6; & x(43)=\delta_8^1; & x(44)=\delta_8^3;\\
x(45)=\delta_8^4; & x(46)=\delta_8^7; & x(47)=\delta_8^6; & x(48)=\delta_8^5; & x(49)=\delta_8^1.
\end{array}$$

设 $x_1(t)$ 的动态方程在向量形式下表为

$$x_1(t+1)=M_1x(t),\quad M_1\in\mathcal{L}_{2\times 8}.$$

由观测数据及前述辨识方法可得

$$\mathrm{Col}_1(M_1)=\begin{cases}\delta_2^1, & 8\text{ 次},\\ \delta_2^2, & 2\text{ 次}.\end{cases} \tag{14.6.3}$$

因此, 设 $\mathrm{Col}_1(M_1)=\delta_2^1$.

对其余 7 列, 有

$$\begin{aligned}
&\mathrm{Col}_2(M_1)=\delta_2^2, \quad 2\text{ 次};\\
&\mathrm{Col}_3(M_1)=\begin{cases}\delta_2^1, & 2\text{ 次},\\ \delta_2^2, & 4\text{ 次};\end{cases}\\
&\mathrm{Col}_4(M_1)=\delta_2^2, \quad 2\text{ 次};\\
&\mathrm{Col}_5(M_1)=\delta_2^1, \quad 9\text{ 次};\\
&\mathrm{Col}_6(M_1)=\begin{cases}\delta_2^1, & 1\text{ 次},\\ \delta_2^2, & 10\text{ 次};\end{cases}\\
&\mathrm{Col}_7(M_1)=\delta_2^2, \quad 6\text{ 次};\\
&\mathrm{Col}_8(M_1)=\delta_2^2, \quad 2\text{ 次}.
\end{aligned} \tag{14.6.4}$$

因此, 可得 M_1 如下:

$$M_1=\delta_2[1\ 2\ 2\ 2\ 1\ 2\ 2\ 2].$$

将 M_1 分为 $M_1=[M_{11}\ \ M_{12}]$, 有 $M_{11}=M_{12}$. $x_1(t)$ 的代数形式为

$$x_1(t+1)=\delta_2[1\ 2\ 2\ 2]x_2(t)x_3(t).$$

返回逻辑形式, 变为

$$x_1(t+1)=x_2(t)\wedge x_3(t).$$

对 $x_2(t)$ 和 $x_3(t)$ 进行相同处理, 可得到它们的逻辑表达式为

$$\begin{cases}x_1(t+1)=x_2(t)\wedge x_3(t),\\ x_2(t+1)=\neg x_1(t),\\ x_3(t+1)=x_1(t)\vee x_2(t).\end{cases} \tag{14.6.5}$$

回查观测数据, 不难检验, 这里有 8 个错误数据. 这种方法看来还是相对鲁棒的.

最后, 值得一提的是, 如果模型已经建好, 而又有新的数据进入, 那可以这样更新模型: 如果第 i 个方程与新数据相符, 则保留它. 否则, 可以增加新的辨识列, 并用新数据构造新的结构矩阵 M_k. 进而更新第 k 个逻辑方程.

14.7 注释与参考

本章考虑从观测数据重构布尔网络模型. 例如, 数据可能来自对癌细胞扩散过程的观察和诊断. 模型重构也称为辨识. 它曾被许多作者研究过, 例如, 文献 [107] 提出一种逆向工程算法来描述基因网络结构; 文献 [18] 提出一种用少量基因来辨识基因模式的方法; 文献 [19] 提出另一种基于矩阵乘法与指纹函数的辨识算法; 文献 [114] 提出一种要求较少平均时间的随机网络搜寻方法等.

我们不直接构造布尔网络的逻辑动态方程, 而是先构造它的代数形式, 然后再返回逻辑形式. 从而给出一般重构方法. 为减少所需数据, 我们又考虑网络图已知的情况, 以及最小入度的实现. 最后考虑齐次布尔网络的情况, 此时辨识与网络大小无关. 对带误差数据的辨识问题也给了一些原则. 本章工作基于文献 [56]. 本章未考虑控制布尔网络的辨识, 这相对比较麻烦. 这方面的工作可参见文献 [55].

第 15 章　非合作博弈

15.1　非合作博弈的数学模型

非合作博弈是目前博弈论研究中最活跃的一个分支. 我们最感兴趣的是有限非合作博弈, 定义如下:

定义 15.1.1　一个 (有限) 正规博弈(normal game) 由三个部分组成: $G=(N,S,C)$, 这里

(i) $N=\{1,2,\cdots,n\}$ 表示玩家, 这里有 n 个玩家;

(ii) $S=\prod_{i=1}^{n} S_i$ 称为局势(profile), 这里

$$S_i=\{s_1^i,\cdots,s_{k_i}^i\},\quad i=1,\cdots,n,$$

是第 i 个玩家的策略(strategy) 集, 它表示第 i 个玩家有 k_i 个策略可供选择. 局势是所有玩家策略集的笛卡儿积.

(iii) $C=(c_1,\cdots,c_n)$, 这里 $c_i: S\to\mathbb{R}$ 是第 i 个玩家的收益函数(payoff function).

为方便计, 策略集通常记为

$$S_i=\{1,2,\cdots,k_i\},\quad i=1,\cdots,n.$$

这里 "有限" 博弈指: (i) 玩家数 $n<\infty$, (ii) 策略数 $|S_i|<\infty$, $i=1,\cdots,n$. 本书今后只讨论有限博弈.

下面给几个例子.

例 15.1.1 (囚徒困境(prisoner's dilemma)[79])　两个小偷被捕. 由于证据不足, 如果两小偷合作 (不招供), 则每人将被判刑 1 年. 如果一个背叛 (即招供), 另一个合作, 则背叛者无罪释放, 合作者被判 9 年. 如果两个均背叛, 则每人将被判刑 6 年. 讨论每个囚徒将如何决策?

这是个典型的博弈问题. (i) $N=\{1,2\}$ 即玩家 1 与玩家 2; (ii) 策略集 $S_1=S_2=\{1,2\}$, 这里 1 代表合作, 2 代表背叛; (iii) 支付函数 c_i. 通常支付函数用一个表来表示. 见表 15.1.1.

表 15.1.1 称为一个支付双矩阵(bi-matrix), 它用不同的行对应玩家 1 不同的策略, 不同的列对应玩家 2 不同的策略. 在每个 i,j 位置给出两个数, 前面是玩家 1 所得, 后面是玩家 2 所得. 例如, 如果玩家 1 取策略 1 (对应第 1 行), 玩家 2 取策略 2 (对应第 2 列), 查 $(1,2)$ 位置可知, 玩家 1 判 9 年, 玩家 2 判 0 年.

表 15.1.1　囚徒困境 (支付双矩阵)

$P_1\backslash P_2$	1	2
1	$-1,\ -1$	$-9,\ 0$
2	$0,\ -9$	$-6,\ -6$

其实, 一个有限博弈的全部信息都表现在这个支付双矩阵中了. 双矩阵虽然被广泛应用, 但是, 当玩家多于两人时它很不方便, 我们试图用单矩阵代替它. 表 15.1.2 是表 15.1.1 的另一种形式.

表 15.1.2　囚徒困境 (支付单矩阵)

支付 \ 局势	11	12	21	22
c_1	-1	-9	0	-6
c_2	-1	0	-9	-6

表 15.1.2 中第 1 行依字典序列出不同的局势 (profile), 第 2 行表示相关局势下玩家 1 所得, 第 3 行相关局势下玩家 2 所得. 下面的例子就看出它的好处.

例 15.1.2　3 个人玩手心手背, 如果某人与其他两人不同, 则他赢 2 分, 其他二人各输 1 分. 记玩家为 1, 2, 3, 则有如下支付矩阵 (单矩阵), 见表 15.1.3.

表 15.1.3　手心手背

支付 \ 局势	111	112	121	122	211	212	221	222
c_1	0	-1	-1	2	2	-1	-1	0
c_2	0	-1	2	-1	-1	2	-1	0
c_3	0	2	-1	-1	-1	-1	2	0

注　*在支付表的第一行, 局势应严格按字典序排列. 即先让最后一个玩家的策略 s_n 从 1 变到 k_n, 然后是倒数第二个玩家的策略 s_{n-1} 从 1 变到 k_{n-1}, 等, 直到第一个玩家的策略 s_1 从 1 变到 k_1. 这样, 才能使支付矩阵有唯一性. 后面将看到, 这种排序在应用上十分有效.*

如果在博弈中, 每种局势下所有收益和为零, 即

$$\sum_{i=1}^{n} c_i(s) = 0, \quad \forall s \in S,$$

则称该博弈为零和博弈. 不难验证, 例 15.1.2 为零和博弈.

15.2 纳什均衡

为方便计, 记

$$S_{-i} := \prod_{j=1, j\neq i}^{n} S_j.$$

定义 15.2.1 给定一个有限博弈 $G=(N,\ S,\ C)$. 一个局势 $s^*=(s_1^*, s_2^*, \cdots, s_n^*)$ 称为一个纳什均衡, 如果

$$c_i(s_i^*, s_{-i}^*) \geqslant c_i(s_i, s_{-i}^*), \quad \forall s_i \in S_i,\ s_{-i}^* \in S_{-i}, \quad i=1,\cdots,n. \tag{15.2.1}$$

纳什均衡表示竞争博弈中一个大家都能接受的策略集合, 因为任何人单独改变策略都会吃亏 (至少不会占便宜), 因此, 没有人有改变策略的动机.

对有限博弈, 很容易从支付矩阵找到纳什均衡点, 我们用一个例子来说明.

例 15.2.1 考虑一个有限博弈 $G=(N,\ S,\ C)$, 这里 $N=\{1,2,3\}$, $S_1=\{1,2,3\}$, $S_2=\{1,2\}$, $S_3=\{1,2,3\}$. 支付矩阵见表 15.2.1.

表 15.2.1 例 15.2.1 支付矩阵

支付 \ 局势	111	112	113	121	122	123	211	212	213
c_1	1	$\underline{2}$	−1	−2	0	1	−2	1	$\underline{1}$
c_2	2	$\underline{3}$	$\underline{4}$	$\underline{3}$	2	1	$\underline{3}$	2	$\underline{2}$
c_3	−2	−1	$\underline{0}$	−4	$\underline{-2}$	−3	−3	−2	$\underline{0}$
支付 \ 局势	221	222	223	311	312	313	321	322	323
c_1	$\underline{1}$	0	$\underline{2}$	$\underline{3}$	$\underline{2}$	$\underline{1}$	−1	$\underline{2}$	−2
c_2	2	$\underline{3}$	1	3	2	$\underline{4}$	$\underline{5}$	$\underline{3}$	1
c_3	−1	−1	$\underline{0}$	$\underline{0}$	−3	−3	−2	$\underline{-1}$	$\underline{-1}$

考虑 c_1, 比较它在 111, 211 以及 311 的值, 这里 $s_{-1} := \prod_{i\neq 1} s_i$ 的策略是一样的. 准确地说, $s_{-1}=s_2\times s_3=11$. 由于 $c_1(111)=1$, $c_1(211)=-2$, 及 $c_1(311)=3$, 故

$$c_1(311)=\max_{s_1\in S_1} c_1(s_1,1,1).$$

那么, 我们在最大值 3 下加一下划线. 类似地, 对每个 $s_{-1}\in S_{-1}$ 寻找最大值

$$\max_{s_1\in S_1} c_1(s_1, s_{-1}), \quad s_{-1}\in S_{-1},$$

对最大值加下划线. 对

$$\max_{s_2\in S_2} c_2(s_2, s_{-2}), \quad s_{-2}\in S_{-2},$$

和

$$\max_{s_3 \in S_3} c_3(s_3, s_{-3}), \quad s_{-3} \in S_{-3},$$

做同样的事.

然后考察支付矩阵的每一列. 如果某一列所有元素都加了下划线, 这一列所对应的局势就是纳什均衡点.

最后, 不难看出, 本例有两个纳什均衡点, 即 (2, 1, 3) 及 (3, 2, 2).

15.3 混合策略

是否所有的有限博弈都有满足 (15.2.1) 的纳什均衡呢? 考察下面这个例子.

例 15.3.1 两人玩石头–剪刀–布(rock-scissors-paper) 游戏. 记策略 $R = 1$, $S = 2$, $P = 3$. 则有如下支付矩阵 (表 15.3.1).

表 15.3.1 石头–剪刀–布

支付 \ 局势	11	12	13	21	22	23	31	32	33
c_1	0	$\underline{1}$	−1	−1	0	$\underline{1}$	$\underline{1}$	−1	0
c_2	0	−1	$\underline{1}$	$\underline{1}$	0	−1	−1	$\underline{1}$	0

用前节办法我们将每种情况下的最佳反应划出. 从表中可以看出, 这个博弈没有定义 15.2.1 所指的纳什均衡.

玩过石头–剪刀–布的人都知道, 应当随机地在 "石头" "剪刀" "布" 中任选一个. 这种策略称为混合策略 (mixed strategy).

定义 15.3.1 给定一个正规博弈.

(i) 记

$$\bar{S}_i = \left\{ \left(r_1^i, \cdots, r_{k_i}^i\right) \middle| r_j^i \geqslant 0, \sum_{j=1}^{k_i} r_j^i = 1 \right\}.$$

$\bar{s}_i = \left(r_1^i, \cdots, r_{k_i}^i\right) \in \bar{S}_i$ 称为一个混合策略, 它表示第 i 个玩家以概率 r_j^i 取策略 s_j^i (简称: 策略 j), $j = 1, \cdots, k_i$.

相应的局势集合记为

$$\bar{S} = \prod_{i=1}^{n} \bar{S}_i.$$

(ii) $\bar{s}^* = (\bar{s}_1^*, \cdots, \bar{s}_n^*)$ 称为一个纳什均衡, 如果支付函数的期望值满足

$$Ec_i(\bar{s}_1^*, \cdots, \bar{s}_n^*) \geqslant Ec_i(\bar{s}_1^*, \cdots, \bar{s}_{i-1}^*, \bar{s}_i, \bar{s}_{i+1}^*, \cdots, \bar{s}_n^*), \quad \bar{s}_i \in \bar{S}_i, \ i = 1, \cdots, n. \tag{15.3.1}$$

由于引进了混合策略, 前面定义的 $s_i \in S_i$ 就称为纯策略(pure strategy). 当然, 纯策略是一种特殊的混合策略.

基于数学期望的线性性, 容易证明: (15.3.1) 等价于下式

$$Ec_i(\bar{s}_1^*, \cdots, \bar{s}_n^*) \geqslant Ec_i(\bar{s}_1^*, \cdots, \bar{s}_{i-1}^*, s_i, \bar{s}_{i+1}^*, \cdots, \bar{s}_n^*), \quad s_i \in S_i, \ i = 1, \cdots, n. \tag{15.3.2}$$

例 15.3.2 回忆例 15.3.1. 如果允许使用混合策略, 不妨设, 第一个玩家的策略为 $(p_1, \ p_2, \ 1-p_1-p_2)$, 第二个玩家的策略为 $(q_1, \ q_2, \ 1-q_1-q_2)$. 于是, 期望值为

$$\begin{aligned}
Ec_1 = & p_1q_2 - p_1(1-q_1-q_2) - p_2q_1 + p_2(1-q_1-q_2) \\
& + (1-p_1-p_2)q_1 - (1-p_1-p_2)q_2, \\
Ec_2 = & -p_1q_2 + p_1(1-q_1-q_2) + p_2q_1 - p_2(1-q_1-q_2) \\
& - (1-p_1-p_2)q_1 + (1-p_1-p_2)q_2.
\end{aligned}$$

注意到纳什均衡是最佳响应函数的解, 计算

$$\frac{\partial Ec_1}{\partial p_1} = q_2 - (1-q_1-q_2) - q_1 + q_2 := 0,$$

可得 $3q_2 = 1$, 于是

$$q_2^* = \frac{1}{3}.$$

同样, 由

$$\frac{\partial Ec_1}{\partial p_2} = 0; \quad \frac{\partial Ec_2}{\partial q_1} = 0; \quad \frac{\partial Ec_2}{\partial q_2} = 0,$$

可分别得到

$$q_1^* = \frac{1}{3}; \quad p_2^* = \frac{1}{3}; \quad p_1^* = \frac{1}{3}.$$

容易检验, $\bar{s}_1^* = \bar{s}_2^* = \left(\frac{1}{3}, \ \frac{1}{3}, \ \frac{1}{3}\right)$ 是纳什均衡点.

为了区别, 由纯策略组成的纳什均衡称为纯纳什均衡.

以下定理是纳什的一个主要贡献.

定理 15.3.1 ([79]) 任何有限博弈总存在纳什均衡, 但它可能是由混合策略构成.

15.4 伪逻辑函数与支付函数

定义 15.4.1 设 $x_i \in \mathcal{D}$, $i = 1, \cdots, n$. 一个映射 $f : \mathcal{D}^n \to \mathbb{R}$, 记作 $f(x_1, \cdots, x_n)$, 称为一个伪布尔函数.

当 x_i 用向量表示时, 即令 $x_i \in \Delta$, 类似于布尔函数, 伪布尔函数也有如下代数表达式.

命题 15.4.1 考查一个伪布尔函数 $f: \mathcal{D}^n \to \mathbb{R}$. 当 x_i 表示为向量形式时, 则存在唯一行向量 $V_f \in \mathbb{R}^{2^n}$, 称为 f 的结构向量, 使得

$$f(x_1, \cdots, x_n) = V_f \ltimes_{i=1}^n x_i. \tag{15.4.1}$$

伪布尔函数有很广泛的应用, 特别是其优化问题曾被广泛讨论[30]. 伪布尔函数的一个直接推广就是伪逻辑函数.

定义 15.4.2 设 $x_i \in \mathcal{D}_{k_i}$, $i = 1, \cdots, n$. 一个映射 $f: \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathbb{R}$, 记作 $f(x_1, \cdots, x_n)$, 称为一个伪逻辑函数. 当 $k_1 = k_2 = \cdots = k_n := k_0$ 时, f 称为 k_0 值伪逻辑函数, 否则称为混合值伪逻辑函数.

同样, 在向量形式下, 令 $x_i \in \Delta_{k_i}$, 伪逻辑函数也有如下代数表达式.

命题 15.4.2 考查一个伪逻辑函数 $f: \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathbb{R}$. 当 x_i 表示为向量形式时, 则存在唯一行向量 $V_f \in \mathbb{R}^k$ (这里 $k = \prod_{i=1}^n k_i$), 称为 f 的结构向量, 使得

$$f(x_1, \cdots, x_n) = V_f \ltimes_{i=1}^n x_i. \tag{15.4.2}$$

回到有限博弈, 考虑策略集 $S_i = \{1, 2, \cdots, k_i\}$. 类似多值逻辑, 只要将 j (第 j 个策略) 用 $\delta_{k_i}^j$ 表示, 则 $S_i \sim \Delta_{k_i}$, $i = 1, \cdots, n$. 于是, 支付函数 $c_i: \prod_{i=1}^n \Delta_{k_i} \to \mathbb{R}$, $i = 1, \cdots, n$, 就成为伪逻辑函数. 记 c_i 的结构向量为 V_i, 则 $(V_1, V_2, \cdots, V_n)$ 唯一确定了一个有限博弈.

命题 15.4.3 $|N| = n$, $|S_i| = k_i$, $i = 1, \cdots, n$ 的有限博弈全体构成一个 nk 维线性空间 $\mathbb{R}^{nk}$, 这里 $k = \prod_{i=1}^n k_i$. 记

$$V := (V_1, V_2, \cdots, V_n) \in \mathbb{R}^{nk}, \tag{15.4.3}$$

这里 V_i 是玩家 i 的支付函数的结构向量.

以后会看到, 有限博弈的线性空间结构非常重要.

15.5 矩阵博弈、凸集与纳什均衡

定义 15.5.1 一个正规博弈 $G = (N, S, C)$ 称为零和博弈, 如果

$$\sum_{i=1}^{n} c_i(s) = 0, \quad \forall s \in S. \tag{15.5.1}$$

下面考虑二人零和博弈. 设

$$S_1 = \{1, 2, \cdots, m\}, \quad S_2 = \{1, 2, \cdots, n\}.$$

分别记 $a_{i,j} = c_1(i,j)$, $b_{i,j} = c_2(i,j)$, 则得

$$b_{i,j} = -a_{i,j}, \quad i = 1, \cdots, m;\ j = 1, \cdots, n.$$

因此, 二人零和博弈也称矩阵博弈, 它由支付矩阵 $A := (a_{i,j})$ 完全决定.

引理 15.5.1 对二人零和博弈, 有

$$\max_{1 \leqslant i \leqslant m} \min_{1 \leqslant j \leqslant n} c(i,j) \leqslant \min_{1 \leqslant j \leqslant n} \max_{1 \leqslant i \leqslant m} c(i,j). \tag{15.5.2}$$

证明 因为

$$\min_{1 \leqslant j \leqslant n} c(i,j) \leqslant c(i,j) \leqslant \max_{1 \leqslant i \leqslant m} c(i,j), \quad i = 1, \cdots, m;\ j = 1, \cdots, n.$$

两边对 i 取最大, 再对 j 取最小, 即得. □

记 $e(i,j) = Ec(i,j)$, 对于混合策略, 同理可得

推论 15.5.1 对零和博弈, 有

$$\max_{\bar{i} \in \bar{S}_1} \min_{\bar{j} \in \bar{S}_2} e(\bar{i}, \bar{j}) \leqslant \min_{\bar{j} \in \bar{S}_2} \max_{\bar{i} \in \bar{S}_1} e(\bar{i}, \bar{j}). \tag{15.5.3}$$

矩阵博弈是一种最基本的, 然而也是最重要的一种博弈. 除了本身的重要性外, 对它的讨论也有助于对一般博弈问题的理解. 相关内容参见参考文献 [12].

定义 15.5.2 在 $\mathbb{R}^n$ (或其他线性空间) 中,

(i) 设 a, $b \in \mathbb{R}^n$, 则

$$\lambda a + (1-\lambda) b, \quad \lambda \in [0, 1]$$

称为 a, b 的凸组合 .

(ii) $S \subset \mathbb{R}^n$ 称为一个凸集, 如果对任意两点 a, $b \in S$, 其凸组合也属于 S.

球、椭球、立方体, 都是 $\mathbb{R}^3$ 上的凸集; 圆、椭圆、三角形、矩形, 都是平面上的凸集.

定理 15.5.1 (凸集分离定理) 设 $S \subset \mathbb{R}^n$ 为一非空闭凸集. $y \in S^c$. 则存在 $0 \neq p \in \mathbb{R}^n$ 以及 $a \in \mathbb{R}$, 使得

$$p^{\mathrm{T}} x \geqslant a > p^{\mathrm{T}} y, \quad \forall x \in S. \tag{15.5.4}$$

证明 因为 S 是非空闭集. 连续函数 $d(x,y)$, $x \in S$, 可达到最小值, 记

$$d(x_0,y) = \min\{d(x,y) \mid x \in S\} > 0.$$

因为 $x_0 \in S$, 对任一 $x \in S$ 有

$$\lambda x + (1-\lambda)x_0 \in S, \quad \lambda \in [0,1].$$

因此,

$$d(x_0,y) \leqslant d(\lambda x + (1-\lambda)x_0, y),$$

即

$$\|x_0 - y\|^2 \leqslant \|\lambda x + (1-\lambda)x_0 - y\|^2.$$

展开即得

$$\lambda\|x - x_0\|^2 + 2(x_0 - y)^{\mathrm{T}}(x - x_0) \geqslant 0.$$

令 $\lambda \to 0^+$ 得

$$(x_0 - y)^{\mathrm{T}}(x - x_0) \geqslant 0, \quad \forall x \in S.$$

令 $p := x_0 - y \neq 0$, $a := p^{\mathrm{T}}x_0$, 则得

$$p^{\mathrm{T}}x \geqslant a, \quad \forall x \in S.$$

另一方面,

$$a - p^{\mathrm{T}}y = p^{\mathrm{T}}(x_0 - y) = \|x_0 - y\|^2 > 0,$$

有

$$p^{\mathrm{T}}x \geqslant a > p^{\mathrm{T}}y, \quad \forall x \in S.$$

□

凸集分离定理的几何意义见图 15.5.1. 它说明: 存在超平面 $H = \{x \in \mathbb{R}^n | p^{\mathrm{T}}x = a\}$ 严格分离 y 与 S.

作为应用, 我们介绍如下命题:

命题 15.5.1 设 $A = (a_{i,j}) \in \mathcal{M}_{m\times n}$. 则下面两情况之一必然成立:

(i) 存在 $y \in \Upsilon_n$, 使得

$$Ay \leqslant 0; \tag{15.5.5}$$

(ii) 存在 $x \in \Upsilon_m$, 使得

$$A^{\mathrm{T}}x > 0. \tag{15.5.6}$$

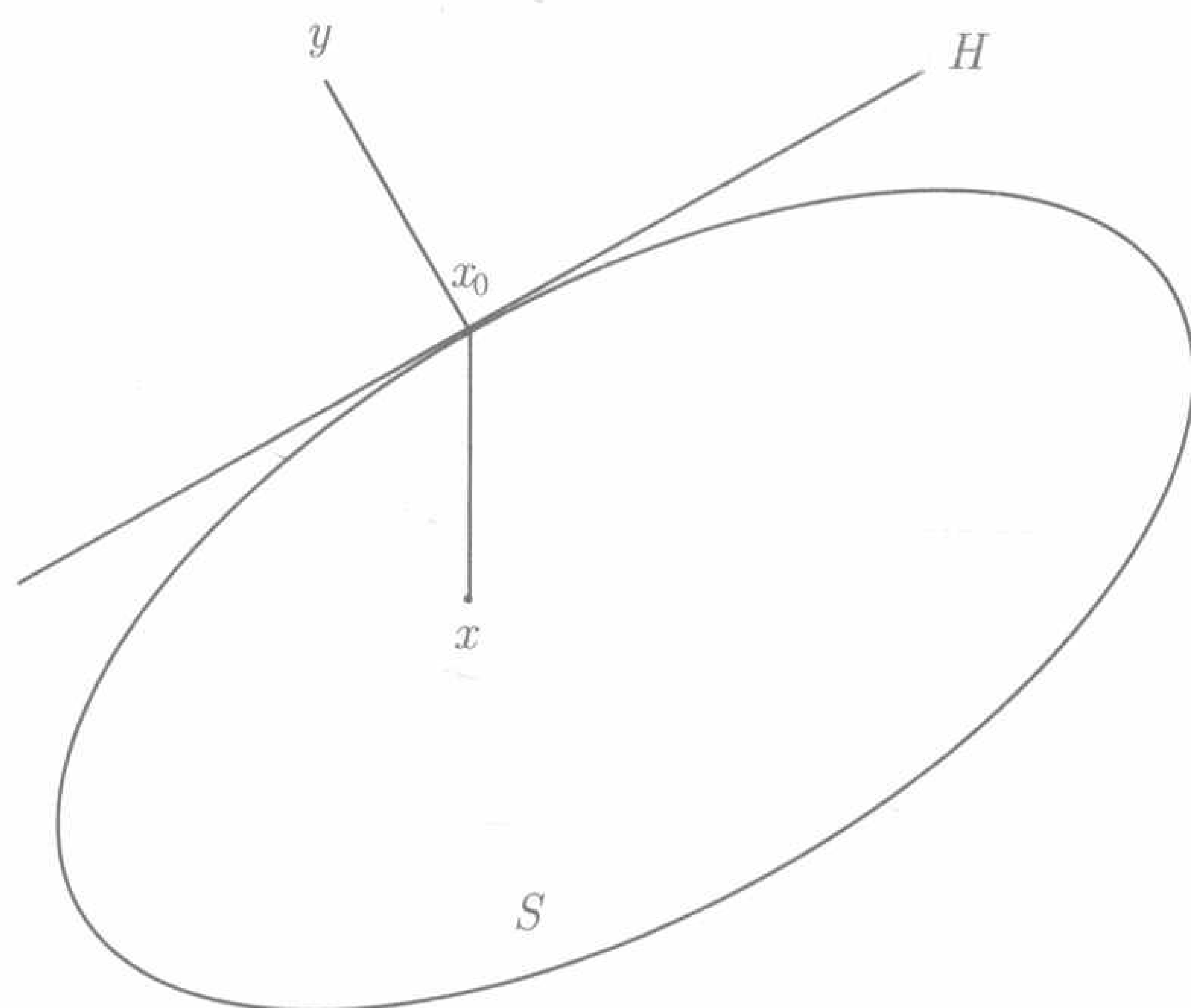


图 15.5.1 凸集分离定理

证明 记 $c_j := \mathrm{Col}_j(A)$, $j = 1, \cdots, n$, 令

$$S = \mathrm{conv}\{c_j,\ j = 1, \cdots, n, \delta^i_m,\ i = 1, \cdots, m\},$$

这里 conv 表示由后面向量生成的凸包. 下面分两种情况讨论:

- $0 \in S$: 则存在 $t \in \Upsilon_{n+m}$, 使得

$$\sum_{i=1}^{n} t_i c_i + \sum_{i=1}^{m} t_{n+i}\delta^i_m = 0.$$

依分量写出, 即得

$$t_1 a_{i,1} + t_2 a_{i,2} + \cdots + t_n a_{i,n} + t_{n+i} = 0, \quad i = 1, \cdots, m.$$

于是有

$$t_1 a_{i,1} + t_2 a_{i,2} + \cdots + t_n a_{i,n} = -t_{n+i} \leqslant 0, \quad i = 1, \cdots, m. \tag{15.5.7}$$

注意: $\sum_{i=1}^{n} t_i > 0$, 否则由 (15.5.7) 可知 $t_i = 0$, $\forall i$, 矛盾. 令

$$y_i = \frac{t_i}{\sum_{i=1}^{n} t_i}, \quad i = 1, \cdots, n.$$

则 $y = (y_1, \cdots, y_n)^{\mathrm{T}} \in \Upsilon_n$, 因此 (15.5.5) 成立.

- $0 \notin S$: 由凸集分离定理知, 存在 $0 \neq p \in \mathbb{R}^m$, 使得

$$p^{\mathrm{T}} c_j > 0, \quad j = 1, \cdots, n. \tag{15.5.8}$$

即

$$A^{\mathrm{T}}p > 0.$$

又有

$$p^{\mathrm{T}}\delta_m^i = p_i > 0, \quad i = 1, \cdots, m.$$

取 $x = \dfrac{p}{\|p\|}$, 则 $x \in \Upsilon_n$, 且满足式 (15.5.6). □

命题 15.5.2 二人零和博弈有纯纳什均衡的充要条件是

$$\max_{1 \leqslant i \leqslant m} \min_{1 \leqslant j \leqslant n} c(i,j) = \min_{1 \leqslant j \leqslant n} \max_{1 \leqslant i \leqslant m} c(i,j). \tag{15.5.9}$$

证明 设 (i^*, j^*) 为纳什均衡点, 则有

$$c(i,j^*) \leqslant c(i^*,j^*) \leqslant c(i^*,j), \quad i = 1,\cdots,m;\ j = 1,\cdots,n. \tag{15.5.10}$$

(必要性) 利用式 (15.5.10), 对 i 取最大, 对 j 取最小, 即得

$$\max_{1 \leqslant i \leqslant m} c(i,j^*) \leqslant c(i^*,j^*) \leqslant \min_{1 \leqslant j \leqslant n} c(i^*,j).$$

由上述不等式与不等式 (15.5.2) 即得等式 (15.5.9).

(充分性) 存在 i^*, j^*, 使得

$$\min_{1 \leqslant j \leqslant n} c(i^*,j) = \max_{1 \leqslant i \leqslant m} \min_{1 \leqslant j \leqslant n} c(i,j);$$

$$\max_{1 \leqslant i \leqslant m} c(i,j^*) = \min_{1 \leqslant j \leqslant n} \max_{1 \leqslant i \leqslant m} c(i,j).$$

由式 (15.5.9) 可得

$$\min_{1 \leqslant j \leqslant n} c(i^*,j) = \max_{1 \leqslant i \leqslant m} c(i,j^*).$$

于是有

$$\begin{aligned} c(i,j^*) &\leqslant \max_{1 \leqslant i \leqslant m} c(i,j^*) = \min_{1 \leqslant j \leqslant n} c(i^*,j) \leqslant c(i^*,j^*) \\ &\leqslant \max_{1 \leqslant i \leqslant m} c(i,j^*) = \min_{1 \leqslant j \leqslant n} c(i^*,j) \leqslant c(i^*,j). \end{aligned}$$

□

对于混合策略, 同理可得

推论 15.5.2 二人零和博弈有混合纳什均衡的充要条件是

$$\max_{\bar{i} \in \bar{S}_1} \min_{\bar{j} \in \bar{S}_2} e(\bar{i},\bar{j}) = \min_{\bar{j} \in \bar{S}_2} \max_{\bar{i} \in \bar{S}_1} e(\bar{i},\bar{j}). \tag{15.5.11}$$

设 $(\bar{i}^*, \bar{j}^*) \in \bar{S}$ 为一 (混合) 纳什均衡, 则称 $\bar{i}^*$ ($\bar{j}^*$) 为玩家 1 (2) 的最优 (混合) 策略.

命题 15.5.3 设 $\bar{i}^*, \bar{j}^*$ 分别为二人零和博弈中玩家 1 和 2 的最优 (混合) 策略. 则 $(\bar{i}^*, \bar{j}^*)$ 为一 (混合) 纳什均衡.

证明 由定义可知, 存在 $\tilde{j}^*$ $(\tilde{i}^*)$ 使 $(\bar{i}^*, \tilde{j}^*)$ ($(\tilde{i}^*, \bar{j}^*)$) 为纳什均衡. 于是有

$$e(\bar{i}, \tilde{j}^*) \leqslant e(\bar{i}^*, \tilde{j}^*) \leqslant e(\bar{i}^*, \bar{j});$$
$$e(\bar{i}, \bar{j}^*) \leqslant e(\tilde{i}^*, \bar{j}^*) \leqslant e(\tilde{i}^*, \bar{j}).$$

因此有

$$e(\tilde{i}^*, \tilde{j}^*) \leqslant e(\bar{i}^*, \tilde{j}^*) \leqslant e(\bar{i}^*, \bar{j}^*) \leqslant e(\tilde{i}^*, \bar{j}^*) \leqslant e(\tilde{i}^*, \tilde{j}^*).$$

这表明

$$e(\bar{i}^*, \tilde{j}^*) = e(\bar{i}^*, \bar{j}^*) = e(\tilde{i}^*, \bar{j}^*).$$

于是, 对任意 $\bar{i} \in \bar{S}_1$, $\bar{j} \in \bar{S}_2$, 有

$$e(\bar{i}, \bar{j}^*) \leqslant e(\tilde{i}^*, \bar{j}^*) = e(\bar{i}^*, \bar{j}^*) = e(\bar{i}^*, \tilde{j}^*) \leqslant e(\bar{i}^*, \bar{j}).$$

□

注 由上述命题可知, 对于二人零和博弈, 如果 $(\tilde{i}^*, \tilde{j}^*)$ 和 $(\bar{i}^*, \bar{j}^*)$ 为两个混合纳什均衡, 则 $(\tilde{i}^*, \bar{j}^*)$ 和 $(\bar{i}^*, \tilde{j}^*)$ 都是混合纳什均衡. 从而可得

$$e(\tilde{i}^*, \tilde{j}^*) = e(\tilde{i}^*, \bar{j}^*) = e(\bar{i}^*, \bar{j}^*).$$

于是, 有一个共同的纳什均衡值.

定理 15.5.2 二人零和博弈有混合纳什均衡的充要条件是

$$\max_{\bar{i} \in \bar{S}_1} \min_{1 \leqslant j \leqslant n} e(\bar{i}, j) = \min_{\bar{j} \in \bar{S}_2} \max_{1 \leqslant i \leqslant m} e(i, \bar{j}). \tag{15.5.12}$$

而且, 这个公共值就是共同的纳什均衡值.

证明 由

$$\min_{1 \leqslant j \leqslant n} e(\bar{i}, j) \geqslant \min_{\bar{j} \in \bar{S}_2} e(\bar{i}, \bar{j});$$

$$\max_{1 \leqslant i \leqslant m} e(i, \bar{j}) \leqslant \max_{\bar{i} \in \bar{S}_1} e(\bar{i}, \bar{j}).$$

可得

$$\max_{\bar{i} \in \bar{S}_1} \min_{1 \leqslant j \leqslant n} e(\bar{i}, j) \geqslant \max_{\bar{i} \in \bar{S}_1} \min_{\bar{j} \in \bar{S}_2} e(\bar{i}, \bar{j}), \tag{15.5.13}$$

$$\min_{\bar{j} \in \bar{S}_2} \max_{1 \leqslant i \leqslant m} e(i, \bar{j}) \leqslant \min_{\bar{j} \in \bar{S}_2} \max_{\bar{i} \in \bar{S}_1} e(\bar{i}, \bar{j}). \tag{15.5.14}$$

另一方面,

$$e(\bar{i},\bar{j}) = \sum_{s=1}^{n} e(\bar{i},s)\bar{j}_s \geqslant \min_{1\leqslant j\leqslant n} e(\bar{i},j);$$

$$e(\bar{i},\bar{j}) = \sum_{s=1}^{m} e(s,\bar{j})\bar{i}_s \leqslant \max_{1\leqslant i\leqslant m} e(i,\bar{j}).$$

可得

$$\max_{\bar{i}\in\bar{S}_1}\min_{\bar{j}\in\bar{S}_2} e(\bar{i},\bar{j}) \geqslant \max_{\bar{i}\in\bar{S}_1}\min_{1\leqslant j\leqslant n} e(\bar{i},j), \tag{15.5.15}$$

$$\min_{\bar{j}\in\bar{S}_2}\max_{\bar{i}\in\bar{S}_1} e(\bar{i},\bar{j}) \leqslant \min_{\bar{j}\in\bar{S}_2}\max_{1\leqslant i\leqslant m} e(i,\bar{j}). \tag{15.5.16}$$

由 (15.5.13) 及 (15.5.15) 可得

$$\max_{\bar{i}\in\bar{S}_1}\min_{\bar{j}\in\bar{S}_2} e(\bar{i},\bar{j}) = \max_{\bar{i}\in\bar{S}_1}\min_{1\leqslant j\leqslant n} e(\bar{i},j).$$

由 (15.5.14) 及 (15.5.16) 可得

$$\min_{\bar{j}\in\bar{S}_2}\max_{\bar{i}\in\bar{S}_1} e(\bar{i},\bar{j}) = \min_{\bar{j}\in\bar{S}_2}\max_{1\leqslant i\leqslant m} e(i,\bar{j}).$$

于是, (15.5.12) 等价于 (15.5.11). 结论显见. □

定理 15.5.3 给定二人零和博弈. $(\bar{i}^*,\bar{j}^*)$ 是混合纳什均衡的充要条件是: 存在实数 φ, 满足

$$e(i,\bar{j}^*) \leqslant \varphi \leqslant e(\bar{i}^*,j), \quad i=1,\cdots,m;\ j=1,\cdots,n. \tag{15.5.17}$$

而且, φ 就是共同的纳什均衡值.

证明 (必要性) 设 $(\bar{x}^*,\bar{y}^*)$ 为纳什均衡点. 取 $\varphi := e(\bar{x}^*,\bar{y}^*)$, 则 (15.5.17) 显然成立.

(充分性) 假设式 (15.5.17) 成立. 有

$$e(\bar{i},\bar{j}^*) = \sum_{s=1}^{m} e(s,\bar{j}^*)\bar{i}_s \leqslant \varphi\sum_{s=1}^{m}\bar{i}_s = \varphi, \quad \forall\bar{i}\in\bar{S}_1;$$

$$e(\bar{i}^*,\bar{j}) = \sum_{s=1}^{n} e(\bar{i}^*,s)\bar{j}_s \geqslant \varphi\sum_{s=1}^{n}\bar{j}_s = \varphi, \quad \forall\bar{j}\in\bar{S}_2.$$

于是有

$$e(\bar{i},\bar{j}^*) \leqslant \varphi \leqslant e(\bar{i}^*,\bar{j}), \quad \forall\bar{i}\in\bar{S}_1,\ \forall\bar{j}\in\bar{S}_2. \tag{15.5.18}$$

令 $\bar{i}=\bar{i}^*$, $\bar{j}=\bar{j}^*$, 则由 (15.5.13) 得 $\varphi = e(\bar{i}^*,\bar{j}^*)$. 代入 (15.5.12) 可知 $(\bar{i}^*,\bar{j}^*)$ 为纳什均衡点. □

15.6 纳什均衡的存在性

定理 15.3.1 说明, 任何有限博弈都存在 (混合) 纳什均衡. (纯纳什均衡可视为混合纳什均衡的一种特例.) 因此, 下面的定理应无新意. 但在矩阵博弈的情况下, 可以给出较简洁的证明, 从而加深我们对纳什均衡的理解.

定理 15.6.1 任何矩阵博弈都存在混合纳什均衡.

证明 记

$$\varphi_1 := \max_{\bar{i}\in\bar{S}_1}\min_{1\leqslant j\leqslant n} e(\bar{i},j);$$
$$\varphi_2 := \min_{\bar{j}\in\bar{S}_2}\max_{1\leqslant i\leqslant m} e(i,\bar{j}).$$

根据定理 15.5.2, 只要证明 $\varphi_1=\varphi_2$ 就行了. 显然 $\varphi_1\leqslant\varphi_2$. 因此, 只要证 $\varphi_1\geqslant\varphi_2$ 即可.

根据命题 15.5.1, 必有下面两情况之一成立.

1) 存在 $\bar{j}\in\Upsilon_n$, 使得

$$e(i,\bar{j})=\sum_{s=1}^{n}c(i,s)\bar{j}_s\leqslant 0,\quad i=1,\cdots,m.$$

于是有

$$\varphi_2=\min_{\bar{j}\in\bar{S}_2}\max_{1\leqslant i\leqslant m} e(i,\bar{j})\leqslant 0. \tag{15.6.1}$$

2) 存在 $\bar{i}\in\Upsilon_m$, 使得

$$e(\bar{i},j)=\sum_{s=1}^{m}c(s,j)\bar{i}_s>0,\quad j=1,\cdots,n.$$

于是有

$$\varphi_1=\max_{\bar{i}\in\bar{S}_1}\min_{1\leqslant j\leqslant n} e(\bar{i},j)>0. \tag{15.6.2}$$

由 (15.6.1) 及 (15.6.2) 可知, $\varphi_1\leqslant 0<\varphi_2$ 不可能成立.

如果将支付函数改为 $\tilde{c}(i,j)=c(i,j)-d$, 其中 d 为任意常数. 重复上述论证即可得到, $\varphi_1-d\leqslant 0<\varphi_2-d$ 不可能成立. 也就是 $\varphi_1\leqslant d<\varphi_2$ 不可能成立. 因 d 是任意的, 故 $\varphi_1<\varphi_2$ 不成立. □

15.7 矩阵博弈的等价性

15.7.1 二人常和博弈

二人常和矩阵博弈的最大特点是, 支付函数满足 $c_1 + c_2 = \text{const}$. 因此, 纳什均衡点满足

$$c_1(\bar{i}, \bar{j}^*) \leqslant c_1(\bar{i}^*, \bar{j}^*) \leqslant c_1(\bar{i}^*, \bar{j}), \quad \forall \bar{i} \in \bar{S}_1, \forall \bar{j} \in \bar{S}_2. \tag{15.7.1}$$

作为特例, 可以用前两节中的极大极小方法讨论纳什均衡解. 实际上, 如果存在 $\alpha > 0, \beta > 0$, 以及 γ, 使得

$$\alpha c_1(s) + \beta c_2(s) + \gamma = 0, \quad \forall s \in S. \tag{15.7.2}$$

那么, 纳什均衡点应满足:

$$\begin{aligned} c_1(\bar{i}, \bar{j}^*) &\leqslant c_1(\bar{i}^*, \bar{j}^*); \\ c_2(\bar{i}^*, \bar{j}) &\leqslant c_2(\bar{i}^*, \bar{j}^*). \end{aligned}$$

第二个方程可变为

$$-\frac{\alpha}{\beta} c_1(\bar{i}^*, \bar{j}) - \frac{\gamma}{\beta} \leqslant -\frac{\alpha}{\beta} c_1(\bar{i}^*, \bar{j}^*) - \frac{\gamma}{\beta}.$$

也就是

$$c_1(\bar{i}^*, \bar{j}^*) \leqslant c_1(\bar{i}^*, \bar{j}).$$

因此, 寻找纳什均衡只要考虑 c_1, 同时, 它满足 (15.7.1). 因此, 前两节的极大极小方法, 以及所有结论, 都可以推广到满足 (15.7.2) 的二人博弈.

当 $\alpha = \beta = 1$ 时, 满足 (15.7.2) 的二人博弈称为二人常和博弈, 它具有特殊的重要性.

15.7.2 等价矩阵博弈

定义 15.7.1 两个矩阵博弈 G^α, G^β 称为等价的, 如果存在实数 $p > 0$ 及 q, 使得它们的基本支付方程满足

$$c^\alpha = pc^\beta + q. \tag{15.7.3}$$

下面的定理是显然的.

定理 15.7.1 设两个矩阵博弈 G^α, G^β 等价. 则 $(\bar{i}^*, \bar{j}^*)$ 是 G^α 的纳什均衡点, 当且仅当 $(\bar{i}^*, \bar{j}^*)$ 是 G^β 的纳什均衡点. 即 G^α 和 G^β 有相同的纳什均衡点集合.

15.8 注释与参考

博弈论也称对策论 (game theory). 各种博弈游戏和朴素的对策思想自从人类社会的出现就产生了, 中国田忌赛马的故事 (公元前 300 余年), 国外如犹太教法典《塔木德》(约公元 500 年) 中的遗产分配问题, 等, 都是有记载的博弈论故事. 而现代博弈论则起源于 20 世纪. 通常认为, 冯 · 诺伊曼 (Von Neumann) 等的论著《博弈论与经济行为》[116] 标志着现代博弈论的诞生.

通常将博弈论分为两类: 非合作 (或曰竞争) 博弈与合作博弈. 纳什 (Nash) 和他提出的纳什均衡是非合作博弈的基础[115], 纳什于 1994 年获诺贝尔经济学奖. 沙普利 (Shapley) 等是合作博弈的代表人物[77], 他于 2012 年获诺贝尔经济学奖.

博弈论的入门书很多, 这里推荐两本: [12], [79].

第16章　演化博弈

16.1　重复博弈的局势演化方程

设 $G=(N,S,C)$ 为一个有限正规博弈. 如果这个博弈被重复多次 (无穷次), 并且, 假定玩家都是理性的, 即每个玩家都会根据已有信息, 设法最大化自己利益, 那么, 这种重复博弈就称为演化博弈. 设 $|N|=n$, $|S_i|=k_i$, $i=1,\cdots,n$. 用 $x_i(t+1)$ 表示玩家 i 在 $t+1$ 次博弈时的策略, 那么, 有以下的演化方程

$$\begin{cases} x_1(t+1)=f_1(x(t),x(t-1),\cdots,x(0)), \\ x_2(t+1)=f_2(x(t),x(t-1),\cdots,x(0)), \\ \qquad\vdots \\ x_n(t+1)=f_n(x(t),x(t-1),\cdots,x(0)), \end{cases} \tag{16.1.1}$$

这里 $x(\tau)\sim(x_1(\tau),\cdots,x_n(\tau))$ 表示 τ 时刻所有策略变量.

最常见的一种演化方程, 其下一时刻策略仅依赖于当下的策略, 这称为马尔可夫决策过程. 对于马尔可夫决策过程, 演化方程变为

$$\begin{cases} x_1(t+1)=f_1(x_1(t),x_2(t),\cdots,x_n(t)) \\ x_2(t+1)=f_2(x_1(t),x_2(t),\cdots,x_n(t)) \\ \qquad\vdots \\ x_n(t+1)=f_n(x_1(t),x_2(t),\cdots,x_n(t)). \end{cases} \tag{16.1.2}$$

(16.1.1) 或 (16.1.2) 通常称为局势演化方程 (strategy profile dynamics). 局势演化方程是由每个玩家的策略演化方程组合而成的. 本章只讨论 (16.1.2) 型演化. 局势演化方程大致可以进一步分为两种:

• 确定型: 这时, $x_i(t)\in\mathcal{D}_{k_i}$, $i=1,\cdots,n$. 在向量形式下, 有 $x_i(t)\in\Delta_{k_i}$. 则在向量形式下有

$$x_i(t+1)=M_i x(t),\quad i=1,\cdots,n, \tag{16.1.3}$$

这里, $x(t)=\ltimes_{i=1}^n x_i(t)$, $M_i\in\mathcal{L}_{k_i\times k}$ 为 f_i 的结构矩阵, $i=1,\cdots,n$. 各式相乘, 则得其代数状态空间方程:

$$x(t+1) = Mx(t), \tag{16.1.4}$$

这里,

$$M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{k\times k}. \tag{16.1.5}$$

• 概率型: 这时, 状态变量用 $x(t) = (r_1, r_2, \cdots, r_k)^{\mathrm{T}} \in \Upsilon_k$ 表示, $P(x(t) = \delta_k^j) = r_j$, $j = 1, \cdots, k$. 这时, (16.1.3) 仍成立, 只是 $\mathrm{Col}_j(M_i) \in \Upsilon_{k_i}$, 它表示 $x(t) = \delta_k^j$ 时 $x_i(t+1)$ 的概率分布. 同样, (16.1.4) 也仍然有效, 但 M 的 (i,j) 元, 记作 $m_{i,j}$, 它表示, $x(t) = \delta_k^j$ 时 $x(t+1) = \delta_k^i$ 的概率. 即

$$m_{ij} = P\left\{x(t+1) = \delta_k^i \mid x(t) = \delta_k^j\right\}, \quad i, j = 1, \cdots, k. \tag{16.1.6}$$

因此, M 是列概率转移矩阵. (16.1.5) 变为

$$M = M_1 * M_2 * \cdots * M_n \in \Upsilon_{k\times k}. \tag{16.1.7}$$

16.2 策略更新规则

在演化博弈中的局势演化方程, 或者说, 每个玩家的策略演化方程, 都是由他们所采用的策略更新规则(strategy updating rule) 来决定的. 目前, 在理论研究中常用的策略更新规则一般是由专家们设计出来的. 下面举出几种常用的.

短视最优响应 (myopic best response adjustment (MBRA))[150]

站在玩家 i 的立场上, 考察其他人在 t 时刻的策略 $s^{-i}(t)$, 选择对付他们的最佳策略, 记作

$$BR_i(t) = \mathrm{argmax}_{s_i\in S_i} c_i(s_i, s^{-i}(t)).$$

这里 BR 为最佳响应 (best response) 策略集合. 那么

- (情况 1) 如果 $x_i(t) \in BR_i(t)$, 则选 $x_i(t+1) = x_i(t)$;
- (情况 2) 如果 $x_i(t) \notin BR_i(t)$, 那么
  - 确定型(记作 MBRA-D): 选择最小下标 j, 使 $s_j \in BR_i(t)$, 然后选定 $x_i(t+1) = s_j$.
  - 概率型 (记作 MBRA-P): 以相同的概率 ($p = 1/|BR_i|$) 任选一个 $s_i \in BR_i$.

对于短视最优响应, 各玩家更新时间很重要. 我们对此做以下划分:

- 时间串联型 (sequential MBRA): 一个时刻只有一个玩家更新策略. 它还可以细分为
  - 周期型串联 (periodical MBRA): 玩家按顺序轮流更新:

$$\begin{cases} x_i(t+1) = f_i(x_1(t), \cdots, x_n(t)), \\ x_j(t+1) = x_j(t), \quad j \neq i; \quad t = kn + (i-1),\ k = 0, 1, 2, \cdots. \end{cases}$$
(16.2.1)

– 随机型串联(stochastic MBRA): 每个玩家以相同的概率 $\left(p = \dfrac{1}{n}\right)$ 被选上更新自己的策略.

- 时间并联型(Parallel MBRA): 所有玩家同时更新他们的策略. 此时, 演化方程即为 (16.1.4).
- 时间级联型(Cascading MBRA): 虽然所有玩家同时更新他们的策略, 但当玩家 j 更新它的策略时, 它知道并使用玩家 i ($i < j$) 的新策略. 即

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)), \\ x_2(t+1) = f_2(x_1(t+1), x_2(t), \cdots, x_n(t)), \\ \qquad \vdots \\ x_n(t+1) = f_n(x_1(t+1), \cdots, x_{n-1}(t+1), x_n(t)). \end{cases} \tag{16.2.2}$$

带参数 $\tau > 0$ 的 Logit 响应 (logit response (LR) with parameter $\tau > 0$)[35]

第 i 个玩家在 $t+1$ 时刻取策略 $s_j \in S_i$ 的概率为

$$P_\tau^i\left(x_i(t+1) = s_j | x(t)\right) = \frac{\exp\left(\dfrac{1}{\tau} c_i(s_j, x^{-i}(t))\right)}{\displaystyle\sum_{s_i \in S_i} \exp\left(\dfrac{1}{\tau} c_i(s_i, x^{-i}(t))\right)}. \tag{16.2.3}$$

无条件模仿 (unconditional imitation (UI))[117]

玩家 i 在所有玩家中选 t 时刻收益最好的玩家, 取其策略为自己下一时刻的策略.

如果最优玩家不唯一:

- 1–型 UI: 取指标最小的. 即设

$$j^* = \min\{\mu | \mu \in \text{argmax}_j c_j(x(t))\}. \tag{16.2.4}$$

则

$$x_i(t+1) = x_{j^*}(t). \tag{16.2.5}$$

- 2–型 UI: 以相同概率取其中任意一个. 即如果

$$\text{argmax}_j c_j(x(t)) := \{j_1^*, \cdots, j_r^*\},$$

则取

$$x_i(t+1) = x_{j_\mu^*}(t), \quad 以概率 \ p_\mu^i = \frac{1}{r}, \quad \mu = 1, \cdots, r. \tag{16.2.6}$$

Fermi 规则 (Fermi's rule (FM))[135, 138]

以等概率任选一个玩家 j. 然后比较 j 与自己的上一次收益, 然后以如下方法决定下一次策略:

$$x_i(t+1) = \begin{cases} x_j(t), & 以概率 \ p_t, \\ x_i(t), & 以概率 \ 1 - p_t, \end{cases} \tag{16.2.7}$$

这里 p_t 由以下 Fermi 函数决定:

$$p_t = \frac{1}{1 + \exp(-\mu(c_j(x(t)) - c_i(x(t))))}.$$

其中, 参数 $\mu > 0$ 可任选. 特别是, 当 $\mu = \infty$ 时可得

$$x_i(t+1) = \begin{cases} x_i(t), & c_i(x(t)) \geqslant c_j(x(t)), \\ x_j(t), & c_i(x(t)) < c_j(x(t)). \end{cases} \tag{16.2.8}$$

16.3 从更新策略到演化方程

本节讨论如何由策略更新规则得到策略及局势演化方程. 可以说, 局势演化方程是完全由策略更新规则所确定的. 在上一节, 我们对几种策略更新规则作了十分详尽的描述, 就是为了使它能唯一确定局势演化方程. 下面, 我们通过具体例子来说明怎样由策略更新规则确定局势演化方程.

例 16.3.1 考察例 15.2.1, 其支付矩阵为表 15.2.1. 我们讨论更新规则如何确定局势演化方程.

考虑策略更新规则为短视最优响应 (MBRA):

对玩家 1, 比较 $c_1(111)$, $c_1(211)$, 与 $c_1(311)$, 因 $c_1(111) = 1$, $c_1(211) = -2$, 与 $c_1(311) = 3$, 故当玩家 2 及玩家 3 都取策略 1 时, 玩家 1 的最佳响应是取策略 3. 也就是说: $f_1(111) = f_1(211) = f_1(311) = 3$. 再比较 $c_1(112) = 2$, $c_1(212) = 1$, 与 $c_1(312) = 2$, 这时, 如果考虑确定型更新 (MBRA-D), 则有 $f_1(112) = f_1(212) = 1$, $f_1(312) = 3$. 如果考虑概率型更新 (MBRA-P), 则有 $f_1(112) = f_1(212) = f_1(312) = \left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right)$. 后面这个记号表示, 取 1 的概率为 $\frac{1}{2}$, 取 3 的概率为 $\frac{1}{2}$. 类此, 就可以得到 f_1 的结构矩阵. 同样, f_2, f_3 的结构矩阵也可得到. 记

$$\begin{cases} x_1(t+1) = f_1(x_1(t), x_2(t), x_3(t)) = M_1 \ltimes_{i=1}^3 x_i(t) := M_1 x(t), \\ x_2(t+1) = f_2(x_1(t), x_2(t), x_3(t)) = M_2 \ltimes_{i=1}^3 x_i(t) := M_2 x(t), \\ x_3(t+1) = f_3(x_1(t), x_2(t), x_3(t)) = M_3 \ltimes_{i=1}^3 x_i(t) := M_3 x(t). \end{cases} \tag{16.3.1}$$

最后可得到局势演化方程

$$x(t+1) = (M_1 * M_2 * M_3)x(t) := Mx(t). \tag{16.3.2}$$

这里

- MBRA-D:

$$\begin{aligned} M_1 &= \delta_3[3,1,2,2,3,2,3,1,2,2,3,2,3,1,2,2,3,2], \\ M_2 &= \delta_2[2,1,1,2,1,1,1,2,1,1,2,1,2,2,1,2,2,1], \\ M_3 &= \delta_3[3,3,3,2,2,2,3,3,3,3,3,3,1,1,1,2,2,2]. \end{aligned} \tag{16.3.3}$$

$$M = \delta_{18}[18,3,9,11,14,8,15,6,9,9,18,9,16,4,7,11,17,8]. \tag{16.3.4}$$

- MBRA-P:

$$\begin{aligned} M_1 = \delta_3 \Big[& 3, 1\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), 2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), \\ & 2,3,2,3,1\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), 2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), \\ & 2,3,2,3,1\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), 2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), 2,3,2 \Big], \\ M_2 = \delta_2 & [2,1,1,2,1,1,1,2,1,1,2,1,2,2,1,2,2,1], \\ M_3 = \delta_3 \Big[& 3,3,3,2,2,2,3,3,3,3,3,3,1,1,1,2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), \\ & 2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right), 1\left(\frac{1}{2}\right) + 2\left(\frac{1}{2}\right) + 3\left(\frac{1}{2}\right) \Big]. \end{aligned}$$

$$\begin{aligned} M = \delta_{18} \Big[& 18, 3\left(\frac{1}{2}\right) + 15\left(\frac{1}{2}\right), 9\left(\frac{1}{2}\right) + 15\left(\frac{1}{2}\right), 11, 14, 8, 15, \\ & 6\left(\frac{1}{2}\right) + 18\left(\frac{1}{2}\right), 9\left(\frac{1}{2}\right) + 15\left(\frac{1}{2}\right), 9, 18, 9, 16, \\ & 4\left(\frac{1}{2}\right) + 10\left(\frac{1}{2}\right), 7\left(\frac{1}{2}\right) + 13\left(\frac{1}{2}\right), 11\left(\frac{1}{2}\right) + 12\left(\frac{1}{2}\right), \\ & 17\left(\frac{1}{2}\right) + 18\left(\frac{1}{2}\right), 8\left(\frac{1}{2}\right) + 9\left(\frac{1}{2}\right) \Big]. \end{aligned}$$

下面考虑更新时间. 实际上, 由前面的操作过程不难看出, (16.3.1) 是时间并联型更新. 对于确定型 (MBRA-D) 演化方程, 我们将其改造成时间级联型更新. 首先有

$$x_1(t+1) = M_1 x(t) := \tilde{M}_1 x(t),$$

这里, $\tilde{M}_1 = M_1$. 其次

$$\begin{aligned}
x_2(t+1) &= M_2 x_1(t+1) x_2(t) x_3(t) \\
&= M_2 M_1 x_1(t) x_2(t) x_3(t) x_2(t) x_3(t) \\
&= M_2 M_1 x_1(t) M_{r,6} x_2(t) x_3(t) \\
&= M_2 M_1 \left(I_3 \otimes M_{r,6}\right) x(t) \\
&:= \tilde{M}_2 x(t),
\end{aligned}$$

于是有

$$\begin{aligned}
\tilde{M}_2 &= M_2 M_1 \left(I_3 \otimes M_{r,6}\right) \\
&= \delta_2[2,1,1,1,2,1,2,1,1,1,2,1,2,1,1,1,2,1].
\end{aligned}$$

同样地, 可得

$$\begin{aligned}
x_3(t+1) &= M_3 x_1(t+1) x_2(t+1) x_3(t) \\
&:= \tilde{M}_3 x(t),
\end{aligned}$$

这里

$$\begin{aligned}
\tilde{M}_3 &= M_3 M_1 \left(I_{18} \otimes \tilde{M}_2\right) M_{r,18} \left(I_6 \otimes M_{r,3}\right) \\
&= \delta_3[2,3,3,3,2,3,2,3,3,3,2,3,2,3,3,3,2,3].
\end{aligned}$$

最后可得局势演化方程为

$$x(t+1) = Lx(t),$$

这里,

$$\begin{aligned}
L &= \tilde{M}_1 * \tilde{M}_2 * \tilde{M}_3 \\
&= \delta_{18}[17,3,9,9,17,9,17,3,9,9,17,9,17,3,9,9,17,9].
\end{aligned}$$

有些策略更新规则一般只用于网络演化博弈. 此时, 有一个网络图. 例如, 无条件模仿. 如果将它用于一般演化博弈, 那么, 一次演化后大家就都一样了, 这种情况没什么意义. 但在网络中, 每一个玩家只在它邻域中选择模仿对象, 这样, 就会出现丰富的演化过程.

再从演化方程的形式看, (16.1.2) 是最常见的一种, 也是本书主要的研究对象. 虽然, 上一节介绍的几种策略更新规则都会导致这种形式的演化方程, 但并不是每一种策略更新规则都如此. 例如, 有一种称为虚拟行为(fictitious play) 的学习博弈规则, 它的策略更新规则是

$$Ex^{-i}(t+1) = \frac{t}{t+1} Ex^{-i}(t) + \frac{1}{t+1} x^{-i}(t). \tag{16.3.5}$$

这里 $Ex^{-i}(t) \in \Upsilon_{k/k_i}$ 表示 $(x_1(t), \cdots, \hat{x}_i(t), \cdots, x_n(t))$ (即, 除 i 外其他玩家的策略的期望值. 而 $x^{-i}(t) \in \Delta_{k/k_i}$ 表示 t 时刻其他玩家真正实现的策略. 然后, 玩家 i 选择对待混合策略 $Ex^{-i}(t+1)$ 的最佳响应

$$\bar{s}_i^* \in \text{argmax}_{\bar{s}_i \in \bar{S}_i} Ec_i \left(\bar{s}_i, Ex^{-i}(t+1)\right)$$

作为自己的新策略:

$$x_i(t+1) = \bar{s}_i^*.$$

虚拟行为是一种很重要的学习博弈规则[112].

16.4　策略的收敛性

与微分方程类似, 局势演化方程最强的稳定性是全局 "渐近稳定". 但由于有限博弈策略的有界 (限) 性, 只要全局收敛就足够了. 仿照微分方程理论, 可以通过构造李雅普诺夫函数的方法来验证收敛性.

定义 16.4.1　给定一个演化博弈 G, 设 $|N| = n$, $|S_i| = k_i$, $i = 1, \cdots, n$, $k := \prod_{i=1}^{n} k_i$.

1) 一个伪逻辑函数 $\psi : \Delta_k \to \mathbb{R}$ 称为 G 的一个李雅普诺夫函数, 如果

$$\psi(x(t+1)) - \psi(x(t)) \geqslant 0, \quad t \geqslant 0, \tag{16.4.1}$$

并且, 如果 $\psi(x(t+1)) = \psi(x(t))$, 则 $x(t+1) = x(t)$.

2) 当使用混合策略时, ψ 应当用其期望值代替, 即, $E\psi : \Upsilon_k \to \mathbb{R}$ 满足

$$E\psi(x(t+1)) - E\psi(x(t)) \geqslant 0, \quad t \geqslant 0, \tag{16.4.2}$$

并且, 如果 $E\psi(x(t+1)) = E\psi(x(t))$, 则 $Ex(t+1) = Ex(t)$.

由定义可推得如下结论.

定理 16.4.1　一个演化博弈, 如果存在一个李雅普诺夫函数, 则它一定会收敛到一个平衡点.

注意, 平衡点未必唯一. 因此, 收敛到哪个平衡点依赖于初值.

设一个演化博弈如前. 其局势演化方程为

$$x(t+1) = Tx(t). \tag{16.4.3}$$

容易验证以下的引理:

引理 16.4.1 一个演化博弈 G 具有李雅普诺夫函数, 当且仅当存在一个行向量 $V_\psi \in \mathbb{R}^k$, 使得

$$V_\psi (T - I_k) \geqslant 0.$$

而且, 如果 $\mathrm{Col}_j (V_\psi (T - I_k)) = 0$, 则可以推出 $\mathrm{Col}_j(T) = \delta_k^j$.

利用这个引理可以得到

定理 16.4.2 设演化博弈 G 具有局势演化方程 (16.4.3), 其中 $T = \delta_k[i_1, i_2, \cdots, i_k]$. G 具有李雅普诺夫函数, 当且仅当

(i) 方程

$$a_{i_j} \geqslant a_j, \quad j = 1, \cdots, k, \tag{16.4.4}$$

有解 $a_j, j = 1, \cdots, k$;

(ii) 如果 $a_{i_j} = a_j$ 则有 $i_j = j$.

下面给一个例子.

例 16.4.1 一个正规博弈 $G = (N, S, C)$, 其中 $N = \{1,2\}$, $S_1 = \{1,2\}$, $S_2 = \{1,2,3\}$, 其支付矩阵见表 16.4.1.

表 16.4.1 例 16.4.1 支付矩阵

支付 \ 局势	11	12	13	21	22	23
c_1	1.1	1.8	2.0	2.2	1.6	3.2
c_2	3.3	2.8	3.1	2.5	3.6	4.1

使用 MBRA, 我们有最佳响应函数如表 16.4.2.

表 16.4.2 例 16.4.1 最佳响应函数

演化函数 \ 局势	11	12	13	21	22	23
f_1	2	1	2	2	1	2
f_2	1	1	1	3	3	3

使用并联 MBRA 则得

$$\begin{aligned} x_1(t+1) &= \delta_2[2,1,2,2,1,2]x(t) := M_1 x(t), \\ x_2(t+1) &= \delta_3[1,1,1,3,3,3]x(t) := M_2 x(t), \end{aligned}$$

于是可得局势演化方程:

$$\begin{aligned} x(t+1) &= Mx(t) = M_1 * M_2 x(t) \\ &= \delta_6[4,1,4,6,3,6]x(t). \end{aligned} \tag{16.4.5}$$

如果选

$$P(x) = \delta_6[2,1,4,5,3,6]x := V_p x. \tag{16.4.6}$$

则有

$$V := V_p(M - I_6) = [3,1,1,1,1,0] \geqslant 0.$$

最后, 可以验证: 如果 $V_j = 0$, 则 $j = 6$. 而当 $j = 6$ 时有 $\mathrm{Col}_j(M) = \delta_6^j$. 因此, $P(x)$ 是该系统的李雅普诺夫函数. 又因平衡点唯一, 该演化博弈全局收敛.

16.5 网络演化博弈的数学模型

我们先给出网络演化博弈的严格定义[65].

定义 16.5.1 一个网络演化博弈由三个元素组成, 记作 $((N,E),G,\Pi)$, 这里

(i) (N,E) 是一个网络图;

(ii) G, 称为基本网络博弈, 当 $(i,j) \in E$ 为网络图的一条边, 则 i 与 j 重复进行基本网络博弈;

(iii) Π 称为策略更新规则.

以下分别讨论这三个组成元素.

■ 网络图

令 $N = \{1,2,\cdots,n\}$ 为顶点, 每个顶点代表一个玩家. $E \subset \{N \times N\}$ 为边. 我们考虑以下两种网络图: (i) 无向图; (ii) 有向图.

定义 16.5.2 设 $i \in N$.

(i) i 的邻域, 记作 $U(i)$, 定义为

$$U(i) := \{j \mid (i,j) \in E, \text{或}(j,i) \in E\} \cup \{i\}. \tag{16.5.1}$$

(ii) i 的 ℓ- 邻域, 记作 $U_\ell(i)$, 可递推地定义为

$$U_\ell(i) := \{j \mid \text{存在 } k \in U_{\ell-1}(i), \text{ 使得 } (j,k) \in E, \text{或}(k,j) \in E\}, \quad \ell = 2,3,\cdots. \tag{16.5.2}$$

定义 16.5.3 一个无向图称为齐次的, 如果它每个顶点的度都一样. 一个有向图称为齐次的, 如果它每个结点的入度和出度分别都一样.

下面给一个例子:

例 16.5.1 设 $N = \{1,2,3,4,5\}$. 它们构成网络图 16.5.1.

(i) 图 16.5.1(a) 是个无向图. 这里

$$U(1) = \{5,1,2\}; \quad U_2(1) = \{4,5,1,2,3\}.$$

(ii) 图 15.5.1(b) 是个有向图. 这里

$$U(1) = \{1,2\}; \quad U_2(1) = \{5,1,2,3\}; \quad U_3(1) = \{4,5,1,2,3\}.$$

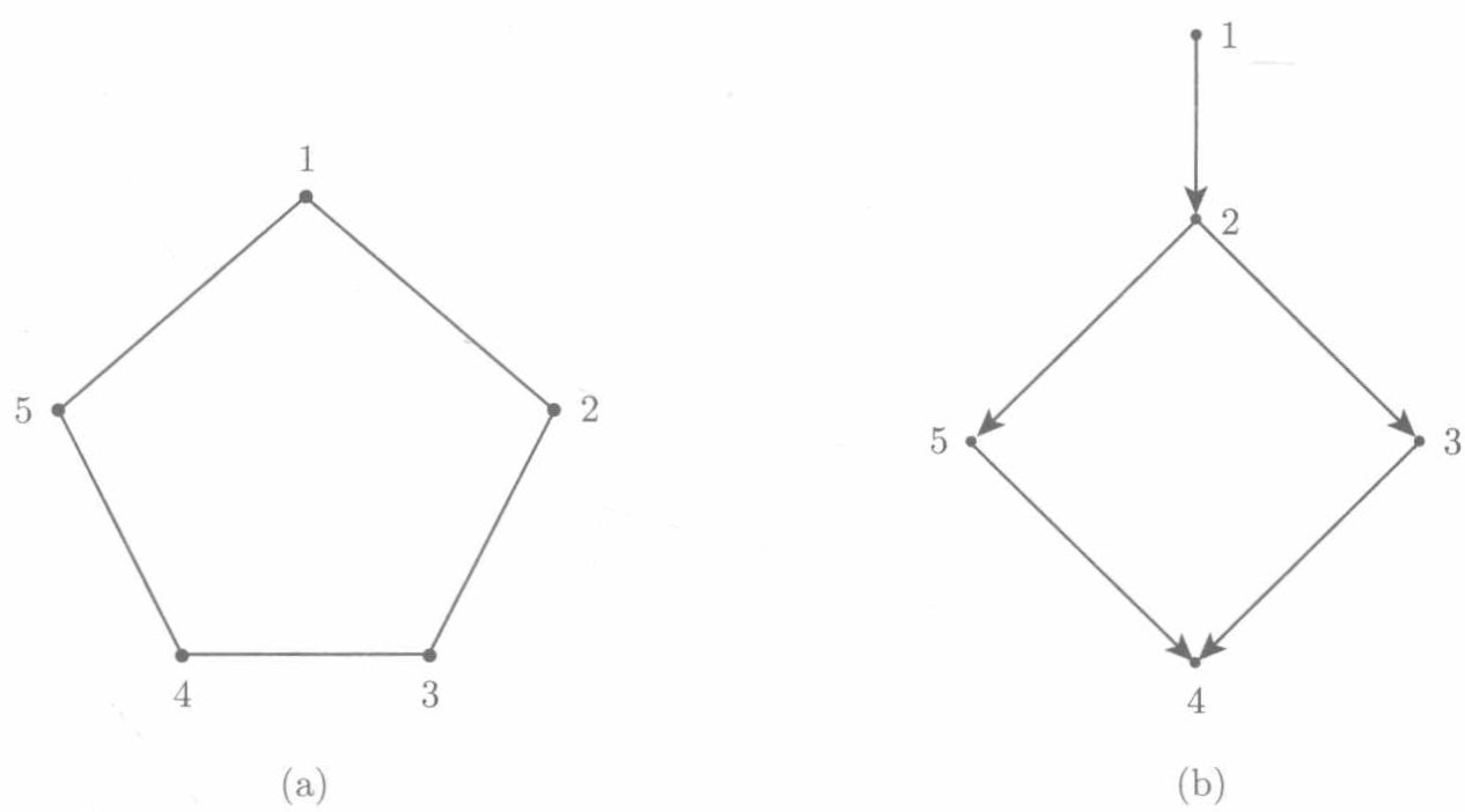


图 16.5.1 网络图

■ 基本网络博弈

定义 16.5.4 基本网络博弈 G 是一个 2 人博弈, 即 $N=\{i,j\}$, 满足

$$S=S_i=S_j=\{1,2,\cdots,k\}.$$

G 称为对称的, 如果

$$c_{i,j}(s_p,s_q)=c_{j,i}(s_q,s_p),\quad \forall s_p,s_q\in S.$$

这里 $c_{i,j}$ 是玩家 i 在与玩家 j 的博弈中的支付. 否则, G 就是非对称的.

注 (i) 如果基本网络博弈 G 是非对称的, 则博弈中玩家 i 和玩家 j 的作用是不同的. 因此, 我们需要有向图. 这时, $(i,j)\in E$ 表示在博弈中前者 i 是第 1 玩家, 后者 j 是第 2 玩家.

(ii) 玩家 i 的总支付可以是总和, 即

$$c_i(t)=\sum_{j\in U(i)\backslash i}c_{i,j}(t),\quad i\in N. \tag{16.5.3}$$

也可以是平均支付, 即

$$c_i(t)=\frac{1}{|U(i)|-1}\sum_{j\in U(i)\backslash i}c_{i,j}(t),\quad i\in N. \tag{16.5.4}$$

基本网络博弈由两个因素决定: (i) k: 策略数; (ii) 类型: 对称与非对称. 因此, 可将其分类如下: (a) $S-k$: 对称 $|S|=k$; (b) $A-k$: 非对称 $|S|=k$.

下面给出一些典型例子. 详细解释可见文献 [23, 123, 134].

例 16.5.2 • $S-2$: 支付双矩阵见表 16.5.1.

表 16.5.1　$S-2$ 博弈

$P_1\backslash P_2$	1	2
1	$(R,\ R)$	$(S,\ T)$
2	$(T,\ S)$	$(P,\ P)$

它包括以下一些著名博弈, 例如

(i) 如果 $2R > T+S > 2P$, 则是囚徒困境;

(ii) 如果 $R=b-c$, $S=b-c$, $T=b$, $P=0$, 且 $2b>c>b>0$, 则成铲雪博弈;

(iii) 如果 $R=\dfrac{1}{2}(v-c)$, $S=v$, $T=0$, $P=\dfrac{v}{2}$, 且 $v<c$, 则成鹰鸽博弈.

- $A-2$: 支付双矩阵见表 16.5.2.

表 16.5.2　$A-2$ 博弈

$P_1\backslash P_2$	1	2
1	$(A,\ B)$	$(C,\ D)$
2	$(E,\ F)$	$(G,\ H)$

它同样包括一些著名博弈, 例如

(1) 如果 $A=H=a$, $B=G=b$,$C=D=E=F=0$, 且 $a>b>0$, 它是性别战;

(ii) 如果 $E>A>C=D>B>0>F$, 且 $G=H=0$, 它是智猪博弈;

(iii) 如果 $A=b$, $B=-b$, $C=b$, $D=-b$, $E=c$, $F=-c$, $G=a$, $H=-a$, 且 $a>b>c>0$, 它是俾斯麦海战;

(iv) 如果 $A=D=F=G-a$, $B=C=E=H=a$, 且 $a\neq 0$, 它是猜硬币游戏.

- $S-3$: 支付双矩阵见表 16.5.3.

表 16.5.3　$S-3$ 博弈

$P_1\backslash P_2$	1	2	3
1	$(A,\ A)$	$(B,\ C)$	$(D,\ E)$
2	$(C,\ B)$	$(F,\ F)$	$(G,\ H)$
3	$(E,\ D)$	$(H,\ G)$	$(I,\ I)$

典型例子包括:

(i) 如果 $A=F=I=0$, $B=E=G=a$, $C=D=H=-a$, 且 $a\neq 0$, 它是石头–剪刀–布游戏;

(ii) 如果 $E=a$, $A=b$, $F=c$, $I=0$, $B=G=H=D=d$, $C=e$, 且 $a>b>c>0>d>c$, 它是 Benoit-Krishna 游戏.

■ **策略更新规则**

本章前面 (第 16.2 节) 介绍过的演化博弈的策略更新规则都可以用于网络演化博弈. 唯一不同之处在于: 在网络演化博弈中, 每一个玩家只能应用其邻域信息来更新自己的策略. 记 $x_i(t)$ 为玩家 i 在 t 时刻的策略, $c_i(t)$ 为玩家 i 在 t 时刻的收益. 那么, 策略更新规则可表示为

$$x_i(t+1) = f_i\left(\{x_j(t), c_j(t) \big| j \in U(i)\}\right), \quad t \geqslant 0, \quad i \in N. \tag{16.5.5}$$

16.6 基本演化方程

考察方程 (16.5.5), 由于 $c_j(t)$ 仅依赖于 $U(j)$ 而 $U(j) \subset U_2(i)$, (16.5.5) 可改写为

$$x_i(t+1) = f_i\left(\{x_j(t) \big| j \in U_2(i)\}\right), \quad t \geqslant 0, \quad i = 1, 2, \cdots, n. \tag{16.6.1}$$

我们称 (16.6.1) 为基本演化方程.

注 *不难看出:*

(i) *网络的整体局势演化方程完全由基本演化方程来确定.*

(ii) *基本演化方程又完全由策略更新规则决定的.*

(iii) *当网络图齐次时, 只有一个基本演化方程. 换言之, 每一点的基本演化方程都一样.*

下面通过几个例子说明如何构造基本演化方程.

例 16.6.1 给定一个网络演化博弈系统, 设其网络图如图 16.5.1(a); 基本网络博弈是 $S-2$ 型, 其中 $R=S=-1$, $T=2$, $P=0$ (即铲雪博弈). 策略更新规则是 1- 型无条件模仿. 用 (mod 5) 形式记顶点, 即 $0=5$, $-1=4$, $6=1$, $7=2$ 等. 于是 $U_2(i) = \{i-2, i-1, i, i+1, i+2\}$. 我们用表 16.6.1 来确定更新策略, 其中, 第一行是 $U_2(i)$ 的局势.

表中首先根据局势, 即 $U_2(i)$ 上各点的策略, 可以得到 $U(i)$ 上各点的收益, 再由收益, 应用策略更新规则, 即得到新策略. 于是, 在向量形式下可得

$$x_i(t+1) = M_f x_{i-2} x_{i-1} x_i x_{i+1} x_{i+2}, \quad i = 1, 2, 3, 4, 5, \tag{16.6.2}$$

这里

$$\begin{aligned} M_f = \delta_2[&1,1,2,2,2,2,2,2,2,2,2,2,2,2,2,2\\ &1,1,2,2,2,2,2,2,2,2,2,2,2,2,2,2]. \end{aligned} \tag{16.6.3}$$

例 16.6.2 给定一个网络演化博弈系统, 设其网络图如图 16.5.1(b); 基本网络博弈是 $A-2$ 型, 其中 $A=H=2$, $B=G=1$, $E=F=C=D=0$ (即性别博弈). 策略更新规则是 Fermi 规则.

表 16.6.1　从支付到策略 (例 16.6.1)

局势	11111	11112	11121	11122	11211	11212	11221	11222
c_{i-1}	-1	-1	-1	-1	-1	-1	-1	-1
c_i	-1	-1	-1	-1	2	2	1	1
c_{i+1}	-1	-1	2	1	-1	-1	1	0
f_i	1	1	2	2	2	2	2	2
局势	12111	12112	12121	12122	12211	12212	12221	12222
c_{i-1}	2	2	2	2	1	1	1	1
c_i	-1	-1	-1	-1	1	1	-2	0
c_{i+1}	-1	-1	2	1	-1	-1	1	0
f_i	2	2	2	2	2	2	2	2
局势	21111	21112	21121	21122	21211	21212	21221	21222
c_{i-1}	-1	-1	-1	-1	-1	-1	-1	-1
c_i	-1	-1	-1	-1	2	2	1	1
c_{i+1}	-1	-1	2	1	-1	-1	1	0
f_i	1	1	2	2	2	2	2	2
局势	22111	22112	22121	22122	22211	22212	22221	22222
c_{i-1}	1	1	1	1	0	0	0	0
c_i	-1	-1	-1	-1	1	1	0	0
c_{i+1}	-1	-1	2	1	-1	-1	1	0
f_i	2	2	2	2	2	2	2	2

注意到基本博弈不对称, 而网络图也不是齐次的, 因此, 只能逐点计算其基本演化方程. 记

$$x_i(t+1) = M_i x_1 x_2 x_3 x_4 x_5 := M_i x, \quad i = 1, 2, 3, 4, 5, \tag{16.6.4}$$

这里 $x = \ltimes_{i=1}^5 x_i$. 结构矩阵 M_i 可通过下面两个步骤计算:

(i) 从局势计算支付. 例如, 设局势为

$$(x_1, x_2, x_3, x_4, x_5) = (1\ 1\ 2\ 2\ 2),$$

那么, 容易算得 (平均支付)

$$\begin{aligned}
c_1 &= 2, \\
c_2 &= \frac{1}{3}(1+0+0) = \frac{1}{3}, \\
c_3 &= \frac{1}{2}(0+1) = \frac{1}{2}, \\
c_4 &= \frac{1}{2}(2+2) = 2, \\
c_5 &= \frac{1}{2}(0+1) = \frac{1}{2}.
\end{aligned}$$

(ii) 比较 c_1 与 c_2, 有 $f_1 = x_1 = 1$. 至于 f_2, 有 3 种选择:

$$j=1 \quad \Rightarrow \quad f_2=x_1=1,$$
$$j=3 \quad \Rightarrow \quad f_2=x_3=2,$$
$$j=5 \quad \Rightarrow \quad f_2=x_5=2.$$

因此, 以概率 1/3 取 $f_2=1$, 概率 2/3 取 $f_2=2$, 记为 $f_2=1(1/3)+2(2/3)$.

类此, 可求得所有 f_i 值, 列表于表 16.6.2. (其中, $a=1(1/2)+2(1/2)$, $b=1(1/3)+2(2/3)$, $c=1(2/3)+2(1/3)$.)

表 16.6.2 从支付到策略 (例 16.6.2)

局势	11111	11112	11121	11122	11211	11212	11221	11222
c_1	2	2	2	2	2	2	2	2
c_2	5/3	1	5/3	1	1	1/3	1	1/3
c_3	3/2	1	1/2	1/2	0	0	1/2	1/2
c_4	1	1/2	0	1	1/2	0	1	2
c_5	3/2	0	1/2	1/2	3/2	0	1/2	1/2
f_1	1	1	1	1	1	1	1	1
f_2	1	1	1	1	1	1	1	b
f_3	1	1	1	a	1	a	a	2
f_4	1	1	1	2	1	1	2	2
f_5	1	1	1	a	1	a	1	2

局势	12111	12112	12121	12122	12211	12212	12221	12222
c_1	0	0	0	0	0	0	0	0
c_2	0	1/3	0	1/3	1/3	2/3	1/3	2/3
c_3	1	1	0	0	1	1	3/2	3/2
c_4	1	1/2	0	1	1/2	0	1	2
c_5	1	1	0	3/2	1	1	0	3/2
f_1	1	2	1	2	2	2	2	2
f_2	c	b	2	2	b	2	2	2
f_3	1	1	1	2	2	2	2	2
f_4	1	a	2	2	a	2	2	2
f_5	1	2	1	2	1	2	2	2

局势	21111	21112	21121	21122	21211	21212	21221	21222
c_1	0	0	0	0	0	0	0	0
c_2	4/3	2/3	4/3	2/3	2/3	0	2/3	0
c_3	3/2	3/2	1/2	1/2	0	0	1/2	1/2
c_4	1	1/2	0	1	1/2	0	1	2
c_5	3/2	0	1/2	1/2	3/2	0	1/2	1/2
f_1	1	1	1	1	1	2	1	2
f_2	1	1	1	1	1	1	1	b
f_3	1	1	1	a	1	2	a	2
f_4	1	1	1	2	1	1	2	2
f_5	1	1	1	a	1	2	a	2

续表

局势	22111	22112	22121	22122	22211	22212	22221	22222
c_1	1	1	1	1	1	1	1	1
c_2	2/3	1	2/3	1	1	4/3	1	4/3
c_3	1	1	0	0	1	1	3/2	3/2
c_4	1	1/2	0	1	1/2	0	1	2
c_5	1	1	0	3/2	1	1	0	3/2
f_1	2	2	2	2	2	2	2	2
f_2	c	2	2	2	2	2	2	2
f_3	1	1	a	2	2	2	2	2
f_4	1	a	2	2	a	2	2	2
f_5	1	2	a	2	1	2	2	2

最后, 可得到

$$\begin{aligned} M_1 = \delta_2[&1,1,1,1,1,1,1,1,1,2,1,2,2,2,2,2\\ &1,1,1,1,1,2,1,2,2,2,2,2,2,2,2,2]. \end{aligned} \tag{16.6.5}$$

$$\begin{aligned} M_2 = \delta_2[&1,1,1,1,1,1,1,1(1/3)+2(2/3),1(2/3)+2(1/3),\\ &1(1/3)+2(2/3),2,2,1(1/3)+2(2/3),2,2,2,1,1,1,1,1,1,1,\\ &1(1/3)+2(2/3),1(2/3)+2(1/3),2,2,2,2,2,2,2]. \end{aligned} \tag{16.6.6}$$

$$\begin{aligned} M_3 = \delta_2[&1,1,1,1(1/2)+2(1/2),1,1(1/2)+2(1/2),1(1/2)+2(1/2),\\ &2,1,1,1,2,2,2,2,2,1,1,1,1(1/2)+2(1/2),1,2,\\ &1(1/2)+2(1/2),2,1,1,1(1/2)+2(1/2),2,2,2,2,2]. \end{aligned} \tag{16.6.7}$$

$$\begin{aligned} M_4 = \delta_2[&1,1,1,2,1,1,2,2,1,1(1/2)+2(1/2),2,2,1(1/2)+2(1/2),2,2,2\\ &1,1,1,2,1,1,2,2,1,1(1/2)+2(1/2),2,2,1(1/2)+2(1/2),2,2,2]. \end{aligned} \tag{16.6.8}$$

$$\begin{aligned} M_5 = \delta_2[&1,1,1,1(1/2)+2(1/2),1,1(1/2)+2(1/2),1,2,1,2,\\ &1,2,1,2,2,2,1,1,1,1(1/2)+2(1/2),1,2,\\ &1(1/2)+2(1/2),2,1,2,1(1/2)+2(1/2),2,1,2,2,2]. \end{aligned} \tag{16.6.9}$$

16.7　从基本演化方程到局势演化方程

本节讨论如何从每个结点的基本演化方程算出网络博弈的整体局势演化方程.

- *确定模型*：

假定经整理 (增加哑变量, 变量顺序重排等) 后有

$$\begin{cases} x_1(t+1) = M_1 x(t), \\ \quad\quad\vdots \\ x_n(t+1) = M_n x(t), \end{cases} \tag{16.7.1}$$

这里 $x(t) = \ltimes_{i=1}^n x_i(t)$, $M_i \in \mathcal{L}_{k\times k^n}$. 那么, 有局势演化方程

$$x(t+1) = Mx(t), \tag{16.7.2}$$

这里

$$M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{k^n\times k^n}. \tag{16.7.3}$$

这里 $*$ 是 Khatri-Rao 积.

例 16.7.1 回忆例 16.6.1. 利用 (16.6.2) 并重排变量可得

$$\begin{aligned} x_1(t+1) &= M_f x_4(t)x_5(t)x_1(t)x_2(t)x_3(t) \\ &= M_f W_{[2^3,2^2]} x(t) := M_1 x(t), \\ x_2(t+1) &= M_f x_5(t)x_1(t)x_2(t)x_3(t)x_4(t) \\ &= M_f W_{[2^4,2]} x(t) := M_2 x(t), \\ x_3(t+1) &= M_f x(t) := M_3 x(t), \\ x_4(t+1) &= M_f x_2(t)x_3(t)x_4(t)x_5(t)x_1(t) \\ &= M_f W_{[2,2^4]} x(t) := M_4 x(t), \\ x_5(t+1) &= M_f x_3(t)x_4(t)x_5(t)x_1(t)x_2(t) \\ &= M_f W_{[2^2,2^3]} x(t) := M_5 x(t). \end{aligned}$$

于是有局势演化方程:

$$x(t+1) = Mx(t), \tag{16.7.4}$$

其中

$$\begin{aligned} M &= M_1 * M_2 * M_3 * M_4 * M_5 \\ &= \delta_{32}[1,20,8,24,15,32,16,32,29,32,32,32,31,32,32,32 \\ &\quad\quad 26,28,32,32,32,32,32,32,30,32,32,32,32,32,32,32]. \end{aligned} \tag{16.7.5}$$

- 概率模型：

实际上, 这时有概率型混合值逻辑动态系统:

$$x(t+1) = M_i x(t), \quad Pr(M_i = M_i^j) = p_i^j, \text{ 且 } \sum_j p_i^j = 1, j = 1,\cdots,s_i;\ i = 1,\cdots,n. \tag{16.7.6}$$

于是有

$$x(t+1) = Mx(t), \tag{16.7.7}$$

这里 $M \in \Upsilon_{k\times k}$ 可计算如下:

$$M = \sum_{j_1=1}^{s_1}\sum_{j_2=1}^{s_2}\cdots\sum_{j_n=1}^{s_n}\left[\left(\prod_{i=1}^{n} p_i^{j_i}\right) M_1^{j_1} * M_2^{j_2} * \cdots * M_n^{j_n}\right]. \tag{16.7.8}$$

我们用下例说明.

例 16.7.2　我们可以利用表 16.6.2 一列一列地计算 M. 例如, 对第 $1\sim 3$ 列, 没有随机项, 显然有

$$\mathrm{Col}_1(M) = \mathrm{Col}_2(M) = \mathrm{Col}_3(M) = \delta_{32}^1.$$

考虑第四列, $\mathrm{Col}_4(M)$ 以概率 1/4 取 δ_{32}^3 或 δ_{32}^4 或 δ_{32}^7 或 δ_{32}^8. 即

$$\begin{aligned}\mathrm{Col}_4(M) = [&0,0,1/4,1/4,0,0,1/4,1/4,0,0,0,0,0,0,0,0,\\ &0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]^{\mathrm{T}}.\end{aligned}$$

记其为

$$\delta_{32}^3\left(\frac{1}{4}\right) + \delta_{32}^4\left(\frac{1}{4}\right) + \delta_{32}^7\left(\frac{1}{4}\right) + \delta_{32}^8\left(\frac{1}{4}\right).$$

逐行计算可得

$$Ex(t+1) = MEx(t), \tag{16.7.9}$$

这里

$$\begin{aligned}M = \delta_{32}[&1,1,1,\alpha,1,\alpha,\beta,\gamma,\mu,\lambda,11,32,\lambda,32,32,32\\ &1,1,1,\alpha,1,22,\alpha,p,\mu,q,r,32,s,32,32,32].\end{aligned} \tag{16.7.10}$$

其中参数为

$$\begin{aligned}
\alpha &= 3\left(\frac{1}{4}\right) + 4\left(\frac{1}{4}\right) + 7\left(\frac{1}{4}\right) + 8\left(\frac{1}{4}\right),\\
\beta &= 3\left(\frac{1}{2}\right) + 7\left(\frac{1}{2}\right),\\
\gamma &= 8\left(\frac{1}{3}\right) + 16\left(\frac{2}{3}\right),\\
\mu &= 1\left(\frac{2}{3}\right) + 9\left(\frac{1}{3}\right),\\
\lambda &= 18\left(\frac{1}{6}\right) + 20\left(\frac{1}{6}\right) + 26\left(\frac{1}{3}\right) + 28\left(\frac{1}{3}\right),\\
p &= 24\left(\frac{1}{3}\right) + 32\left(\frac{2}{3}\right),
\end{aligned}$$

$$q=26\left(\frac{1}{2}\right)+28\left(\frac{1}{2}\right),$$
$$r=27\left(\frac{1}{4}\right)+28\left(\frac{1}{4}\right)+31\left(\frac{1}{4}\right)+32\left(\frac{1}{4}\right),$$
$$s=29\left(\frac{1}{2}\right)+31\left(\frac{1}{2}\right).$$

其实, 如果将 (16.7.8) 写成期望形式, 即

$$Ex_i(t+1)=\sum_{j=1}^{s_i}M_i^jEx(t):=M_iEx(t),\quad i=1,\cdots,n,\tag{16.7.11}$$

这里 $M_i\in\Upsilon_{k_i\times k}$, 那么, 不难验证, 公式 (16.7.3) 仍然成立.

16.8 网络演化博弈的控制

定义 16.8.1 设 $((N,E),G,\Pi)$ 为一网络演化博弈, 这里 $N=U\cup Z$ 为顶点 N 的一个分割. 称 $((U\cup Z),E),G,\Pi)$ 为一个控制网络演化博弈, 如果玩家 $u\in U$ 的策略可以任选. 对于控制网络演化博弈 $z\in Z$ 称为状态, $u\in U$ 称为控制.

实际上, 一旦局势演化方程得到了, 那么, 网络演化博弈的控制问题就与混合值逻辑网络的控制一样了. 现在设 $|N|=n+m$, 其中 $|Z|=n$, $|U|=m$. 对于每一个 $i\in Z$, 设 $x_i\in\Delta_{k_i}$, 对于每一个 $j\in U$, 设 $u_j\in\Delta_{\ell_j}$, 然后记 $k=\prod_{i=1}^n k_i$, $\ell=\prod_{j=1}^m \ell_j$.

根据 G 和 Π, 显然可以得到 $i\in Z$ 的策略演化方程, 记为

$$x_i(t+1)=M_iu(t)x(t),\quad i=1,\cdots,n,\tag{16.8.1}$$

这里, $u(t)=\ltimes_{j=1}^m u_j(t)$, $x(t)=\ltimes_{j=1}^n x_j(t)$. 于是, 可得带控制的局势演化方程

$$x(t+1)=Mu(t)x(t),\tag{16.8.2}$$

这里 $M=M_1*\cdots*M_n\in\mathcal{L}_{k\times k\ell}$.

对于混合策略的情况, 则演化方程为概率型的, 即 $x_i\in\Upsilon_{k_i}$, $i=1,\cdots,n$, $u_j\in\Upsilon_{\ell_j}$, $j=1,\cdots,m$. 局势演化方程变为

$$Ex(t+1)=MEu(t)x(t),\tag{16.8.3}$$

这里 $M=M_1*\cdots*M_n\in\Upsilon_{k\times k\ell}$.

例 16.8.1 回忆例 16.6.1. 假定玩家 2 和 4 为控制, 其余为状态. 即

$$u_1=x_2,\quad u_2=x_4,\quad z_1=x_1,\quad z_2=x_3,\quad z_3=x_5.$$

利用例 16.6.1 中的已知结果, 有

$$
\begin{aligned}
x_1(t+1) &= M_f x_4(t)x_5(t)x_1(t)x_2(t)x_3(t) \\
&= M_f W_{[2^3,2^2]} x_1(t)x_2(t)x_3(t)x_4(t)x_5(t) \\
&= M_f W_{[2^3,2^2]} W_{[2,2^3]} x_4(t)x_1(t)x_2(t)x_3(t)x_5(t) \\
&= M_f W_{[2^3,2^2]} W_{[2,2^3]} W_{[2,2^2]} u_1(t)u_2(t)z_1(t)z_2(t)z_3(t) \\
&:= L_1 u(t)z(t),
\end{aligned}
$$

这里 $L_1 = M_f W_{[2^3,2^2]} W_{[2,2^3]} W_{[2,2^2]}$, $u(t) = u_1(t)u_2(t)$, $z(t) = z_1(t)z_2(t)z_3(t)$. 类似地, 可以算得

$$
z_i(t+1) = L_i u(t)z(t), \quad i = 1,2,3, \tag{16.8.4}
$$

这里

$$
\begin{aligned}
L_1 &= M_f W_{[2^3,2^2]} W_{[2,2^3]} W_{[2,2^2]} \\
&= \delta_2[1,2,1,2,2,2,2,2,1,2,1,2,2,2,2,2, \\
&\qquad 2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2]; \\
L_2 &= M_f W_{[2,2^3]} W_{[2,2^2]} \\
&= \delta_2[1,1,2,2,1,1,2,2,2,2,2,2,2,2,2,2, \\
&\qquad 2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2]; \\
L_3 &= M_f W_{[2^2,2^3]} W_{[2,2^3]} W_{[2,2^2]} \\
&= \delta_2[1,2,1,2,2,2,2,2,2,2,2,2,2,2,2,2, \\
&\qquad 1,2,1,2,2,2,2,2,2,2,2,2,2,2,2,2].
\end{aligned}
$$

最后, 控制网络演化博弈的局势演化方程为

$$
z(t+1) = Lu(t)z(t), \tag{16.8.5}
$$

这里

$$
\begin{aligned}
L &= L_1 * L_2 * L_3 \\
&= \delta_8[1,6,3,8,6,6,8,8,4,8,4,8,8,8,8,8, \\
&\qquad 7,8,7,8,8,8,8,8,8,8,8,8,8,8,8,8].
\end{aligned}
$$

有了控制网络演化博弈的局势演化方程, 则所有相应的控制问题都可以进行讨论了.

16.9　演化策略的稳定性

演化稳定策略的概念最初来自生物系统. 按文献 [134] 的定义: 考虑一个演化博弈群体. 一个策略称为演化稳定策略, 如果群体成员均采用这一策略, 那么, 一个变异在自然选择下无法侵入该群体. 我们通过下面这个例子来理解它.

例 16.9.1 ([134]) 考虑鹰鸽博弈(hawk-dove game). 这个博弈是对称的, 支付矩阵见表 16.9.1, 这里 $E(X,Y)$ 表示取策略 X 的玩家对取策略 Y 的玩家时的所得, $X,Y \in \{H,D\}$.

表 16.9.1 支付矩阵 (鹰鸽博弈)

$P_1\backslash P_2$	H	D
H	$E(H,H)$	$E(H,D)$
D	$E(D,H)$	$E(D,D)$

设 p 为群体中取策略 H 的概率, $W(H)$ 和 $W(D)$ 分别为策略 H 与 D 的适应度. 所有个体初始适应度均为 W_0. 则在一次博弈中, 适应度为

$$\begin{cases} W(H) = W_0 + pE(H,H) + (1-p)E(H,D), \\ W(D) = W_0 + pE(D,H) + (1-p)E(D,D). \end{cases} \tag{16.9.1}$$

假如个体是无性繁殖, 后代数目正比于适应度. 在后代中取策略 H 的概率 p' 为

$$p' = pW(H)/\overline{W}, \tag{16.9.2}$$

这里 $\overline{W} = pW(H) + (1-p)W(D)$.

现在假定 H 是演化稳定策略而 D 是变异, 那么, $(1-p)$ 应当非常小. 因为 H 是稳定的, 则 $W(H) > W(D)$. 于是有

$$\begin{aligned} &\text{或者} \quad E(H,H) > E(D,H), \\ &\text{或者} \quad E(H,H) = E(D,H) \text{ 且 } E(H,D) > E(D,D). \end{aligned} \tag{16.9.3}$$

关系式 (16.9.3) “可以作为演化稳定策略的一个定量的定义, 但它只能用于无限群体、无性繁殖和成对博弈的情况.”[133].

下面考虑网络演化博弈的演化稳定策略. 设有一网络演化博弈, 其网络图为图 16.9.1, 演化稳定策略由 (16.9.3) 定义. 设所有玩家都采用策略 H, 仅在结点 O 出现变异.

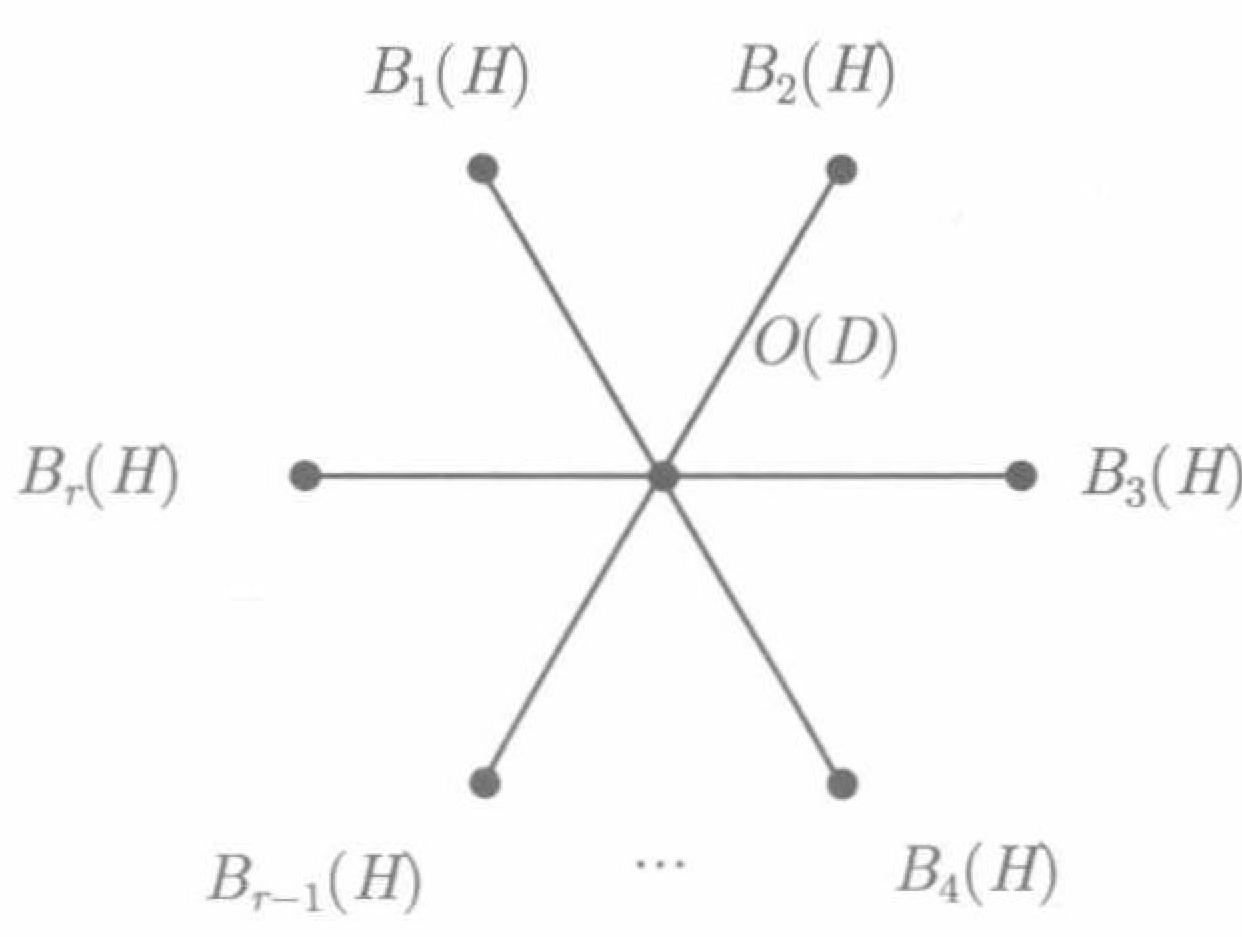


图 16.9.1 O 点变异的网络

回到 (16.9.3), 看它对这个网络演化博弈是否成立. 由于变异是小概率事件, 假定 $U_2(O)$ 中没有其他变异. 假定 $W_0 = 0$, 设 $U(O) = \{O, B_1, \cdots, B_r\}$ 并且 $|U(B_i)| - 1 = \ell_i$, $i = 1, \cdots, r$.

$$
\begin{aligned}
c_i(B_i) &= \frac{1}{\ell_i}\left[(\ell_i - 1)E(H,H) + E(H,D)\right], \quad i = 1, \cdots, r, \\
c_0(O) &= E(D,H).
\end{aligned}
$$

假定策略更新规则为 Π-I, 即选择最佳邻域策略为下一步策略. 不难验证, 即使假定 $E(H,H) > E(D,H)$, 仍然不能保证 $c_i(B_i) > c_0(O)$. 因此, 即使不断演化下去, 变异也无法消去.

再说, 对于网络演化博弈, 由 (16.9.2) 所定义的策略更新规则也不合适. 因为在网络演化博弈中, 每一个玩家只能得到他邻居的信息, 并依此更新自己的策略.

因此, 我们对网络演化博弈给出如下的定义, 不难看出, 它与文献 [134] 中的一般定义相符.

设 $x = (x_1, \cdots, x_n) \in S$, $y = (y_1, \cdots, y_n) \in S$ 为两个局势, 这里 x_i, $y_i \in \Delta_k$, $i = 1, \cdots, n$. 则

$$
\|x - y\| := \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{k}\left|(x_i)_j - (y_i)_j\right|. \tag{16.9.4}
$$

定义 16.9.1　1) 给定一个网络演化博弈, 策略 $\xi \in S$ 称为演化稳定策略, 如果存在 $\mu \geqslant 1$, 使当初始策略 y_0 满足

$$
\|y_0 - x_0\| \leqslant \mu, \tag{16.9.5}
$$

则成立

$$
\lim_{t \to \infty} y(t, y_0) = x_0, \tag{16.9.6}
$$

这里 $x_0 = \xi^n$. 并且, ξ 称为具有水平 μ 的演化稳定策略.

2) 如果对一个 $i \in N$, 在 $U_k(i)$ 内容许一个不超过 μ 的变异. 准确地说,

$$
\|y_0 - x_0\| \leqslant \mu, \quad y_0 \in U_k, \forall i \in N, \tag{16.9.7}
$$

有 (16.9.6), 那么, ξ 称为具有水平 $\mu/[k]$ 的演化稳定策略.

注　(i) 当群体个数 n 有限时, (16.9.6) 可改写如下: 存在一个 $T > 0$ 使得

$$
y(t, y_0) = x_0, \quad t \geqslant T. \tag{16.9.8}
$$

(ii) 由定义式 (16.9.4) 可知, 定义 16.9.1 表示, 至少一个变异不会影响演化结果. 所以通常取 μ 为正整数.

(iii) 显然 μ 可以用来量测稳定性程度: μ 越大则稳定性越高.

因为策略局势的动态方程刻画了整个网络的演化, 它可以用来检验一个策略是否为稳定演化策略. 下面通过一个例子来说明这一点.

例 16.9.2 设网络图为 S_7 (七个玩家联成一个环), 以 (mod 7) 记玩家. 则玩家 i 的邻域为 $U(i)=\{i-1,i,i+1\}$, 其 2 次邻域为 $U_2(i)=\{i-2,i-1,i,i+1,i+2\}$. 设基本网络博弈为囚徒困境. 即, $S_0=\{1,2\}$, 这里 1 代表 "合作" 而 2 代表 "背叛". 支付双矩阵见表 16.9.2.

表 16.9.2 囚徒困境支付双矩阵

$P_1\backslash P_2$	1	2
1	$(R,\ R)$	$(S,\ T)$
2	$(T,\ S)$	$(P,\ P)$

设 $P=-6, R=-5, S=-5, T=-3$, 那么, 策略局势函数可由表 16.9.3 给出. 这里第一行是 $(x_{i-2},x_{i-1},x_i,x_{i+1},x_{i+2})$ 的策略.

表 16.9.3 从支付到下一个策略

局势	11111	11112	11121	11122	11211	11212	11221	11222
$c_{i-1}(t)$	−5	−5	−5	−5	−5	−5	−5	−5
$c_i(t)$	−5	−5	−5	−5	−3	−3	−4.5	−4.5
$c_{i+1}(t)$	−5	−5	−3	−4.5	−5	−5	−4.5	−6
$x_i(t+1)$	1	1	2	2	2	2	2	2
局势	12111	12112	12121	12122	12211	12212	12221	12222
$c_{i-1}(t)$	−3	−3	−3	−3	−4.5	−4.5	−4.5	−4.5
$c_i(t)$	−5	−5	−5	−5	−4.5	− 4.5	−6	−6
$c_{i+1}(t)$	−5	−5	−3	−4.5	−5	−5	−4.5	−6
$x_i(t+1)$	2	2	2	2	2	2	2	2
局势	21111	21112	21121	21122	21211	21212	21221	21222
$c_{i-1}(t)$	−5	−5	−5	−5	−5	−5	−5	−5
$c_i(t)$	−5	−5	−5	−5	−3	−3	−4.5	−4.5
$c_{i+1}(t)$	−5	− 5	−3	−4.5	−5	−5	−4.5	−6
$x_i(t+1)$	1	1	2	2	2	2	2	2
局势	22111	22112	22121	22122	22211	22212	22221	22222
$c_{i-1}(t)$	−4.5	−4.5	−4.5	−4.5	−6	−6	−6	−6
$c_i(t)$	−5	−5	−5	−5	−4.5	−4.5	−6	−6
$c_{i+1}(t)$	−5	−5	−3	−4.5	−5	−5	−4.5	−6
$x_i(t+1)$	2	2	2	2	2	2	2	2

将表 16.9.3 中的 $x_i(t+1)$ 放到一起, 则得基本演化方程如下:

$$x_i(t+1)=M\ltimes_{j=-2}^{2}x_{i+j}(t), \tag{16.9.9}$$

这里, 结构矩阵为

$$
\begin{aligned}
M = \delta_2[&1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2 \\
&1\ 1\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2].
\end{aligned}
\tag{16.9.10}
$$

利用基本演化方程 (16.9.9)–(16.9.10), 不难算得策略局势动态方程如下:

$$
x(t+1) = Lx(t), \tag{16.9.11}
$$

这里 $x(t) = \ltimes_{i=1}^{7} x_i(t)$, 且

$$
\begin{aligned}
L = \delta_{128} & \\
[1 \quad 68 \quad 8 \quad 72 \quad 15 \quad 80 \quad 16 \quad 80 \quad 29 \quad 96 \quad 32 \quad 96 \\
31 \quad 96 \quad 32 \quad 96 \quad 57 \quad 124 \quad 64 \quad 128 \quad 63 \quad 128 \quad 64 \quad 128 \\
61 \quad 128 \quad 64 \quad 128 \quad 63 \quad 128 \quad 64 \quad 128 \quad 113 \quad 116 \quad 120 \quad 120 \\
127 \quad 128 \quad 128 \quad 128 \quad 125 \quad 128 \quad 128 \quad 128 \quad 127 \quad 128 \quad 128 \quad 128 \\
121 \quad 124 \quad 128 \quad 128 \quad 127 \quad 128 \quad 128 \quad 128 \quad 125 \quad 128 \quad 128 \quad 128 \\
127 \quad 128 \quad 128 \quad 128 \quad 98 \quad 100 \quad 104 \quad 104 \quad 112 \quad 112 \quad 112 \quad 112 \\
126 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 122 \quad 124 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 126 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
114 \quad 116 \quad 120 \quad 120 \quad 128 \quad 128 \quad 128 \quad 128 \quad 126 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 122 \quad 124 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
126 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128].&
\end{aligned}
$$

于是可直接检验, $k \geqslant 3$ 时,

$$
\begin{aligned}
L^k = \delta_{128} & \\
[1 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \\
128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128 \quad 128].&
\end{aligned}
$$

也就是说, 当 $x(0) = \delta_{128}^1$ 时, 局势收敛于 $x(\infty) = x(3) = \delta_{128}^1 \sim (1,1,1,1,1,1,1)$. 而当初态 $x(0) \neq \delta_{128}^1$ 时, 局势收敛于 $\delta_{128}^{128} \sim (2,2,2,2,2,2,2)$.

我们得出结论: $\xi = \delta_2^2 \sim 2$, 即策略 2, 是演化稳定策略. 而且, 我们可以选择 $\mu = 6$, 即当 $|y_0 - x_0| \leqslant \mu$ 时 (这里, $x_0 = \xi^7$), 只要令 $T = 3$, 则 (16.9.8) 成立. 因此, 稳定水平为 6.

考虑另一个平衡点 $\eta = \delta_2^1 \sim 1$. 显然, η 不是演化稳定策略, 因为对任何变异 $|y_0 - \eta^7| \geqslant 1$, 都有 $y(t, y_0) \to \xi$. 这表明任何一个变异都会侵入群体, 最后使群体完全改变.

在上述例子中, 设 $H = \delta_2^2 \sim 2$ 以及 $D = \delta_2^1 \sim 1$. 那么, 不难看出

$$E(H, H) = -6 < E(D, H) = -5.$$

即, (16.9.3) 不再成立. 因此, 对有限集上的网络演化博弈, (16.9.3) 不是演化稳定策略的必要条件. 实际上, 可以证明, 它也不是充分的.

16.10 注释与参考

演化博弈最早是由生物学家提出来的. 它讨论在重复博弈中如何做决策以及由此引起的结果. “虽然博弈论最初是为研究经济行为而设计的, 但结果却更好地应用到了生物学研究之中.”[134] 这是因为经济行为中的自利原则 (理性) 在生物学中被达尔文的适应度所代替了, 而后者更具客观性. 本章的主要内容基于文献 [63], [65].

在生物学或经济行为中有一种演化博弈, 参与者是一个很大的群体. 每个玩家只与其邻域中的玩家进行博弈. 这时群体间的拓扑结构是很重要的, 它用一个网络图来刻画. 这种博弈称为网络演化博弈. 通常, 在这种博弈中, 每个玩家只能得到他邻域的信息来更新他的策略[63, 65].

第17章　势　博　弈

17.1　势函数与势博弈

定义 17.1.1　设 $G=(N,S,C)$ 为一有限博弈. $|N|=n$, $|S_i|=k_i$, $i=1,\cdots,n$, $\prod_{i=1}^{n} k_i = k$.

(i) G 称为一个泛势博弈(ordinal potential game), 如果存在一个函数 $P:S\to\mathbb{R}$, 称作势函数, 使得对每个 i 和每个 $s^{-i}\in S^{-i}$ 均成立

$$c_i(x_i,s^{-i})-c_i(y_i,s^{-i})>0 \Leftrightarrow P(x_i,s^{-i})-P(y_i,s^{-i})>0, \quad \forall x_i,y_i\in S_i. \tag{17.1.1}$$

(ii) G 称为一个加权势博弈(weighted potential game), 如果存在一组正数 $\{w_i>0 \mid i=1,\cdots,n\}$, 称为权重, 和一个函数 $P:S\to\mathbb{R}$, 称为加权势函数, 使得对每个 i, 每一组 x_i, $y_i\in S_i$, 和每一个 $s^{-i}\in S^{-i}$ 均成立

$$c_i(x_i,s^{-i})-c_i(y_i,s^{-i})=w_i\left[P(x_i,s^{-i})-P(y_i,s^{-i})\right]. \tag{17.1.2}$$

(iii) G 称为一个 (纯) 势博弈(pure potential game), 如果 G 是一个加权势博弈, 且所有权重均为 1, 即 $w_i=1$, $\forall i$. 相关的 P 称为 (纯) 势函数.

注意, 我们显然有如下蕴涵关系:

$$\text{势博弈} \Rightarrow \text{加权势博弈} \Rightarrow \text{泛势博弈}.$$

下面是势博弈的一些主要性质:

定理 17.1.1 ([112])　如果 G 是势博弈, 那么, P 在容许一个常数差的意义下唯一. 换言之, 如果 P_1 和 P_2 为 G 的两个势函数, 则 $P_1-P_2=c_0\in\mathbb{R}$.

定理 17.1.2 ([112])　如果 G 是势博弈, P 是 G 的势函数, s^* 为势函数的一个极大值点. 那么, s^* 是 G 的一个纳什均衡点.

下面的推论是显然的.

推论 17.1.1　如果 G 是有限势博弈, 由它依串联 MBRA 更新方式形成演化博弈, 则该演化博弈收敛于一个纳什均衡点.

证明　根据势博弈的定义, 串联 MBRA 的每一步更新都会让势函数增加. 但所有局势是有限的, 在有限步后一定会达到极大值点. □

注意, 极大值点未必是最大值点, 它只是保证局部最大. 如果极大值点唯一, 则成最大值点.

熟知, 无论是力学或电场中的势函数对闭路增量为零. 下面的定理显示了博弈中势函数的类似性质. 它也被用来检验一个博弈是否是势博弈.

定理 17.1.3([112]) 一个博弈 G 是势博弈, 当且仅当对每一对 $i,j \in N$, 选择任何一个 $a \in S^{-\{i,j\}}$, 一对 $x_i, y_i \in S_i$, 和一对 $x_j, y_j \in S_j$, 均有

$$c_i(B) - c_i(A) + c_j(C) - c_j(B) + c_i(D) - c_i(C) + c_j(A) - c_j(D) = 0, \tag{17.1.3}$$

这里 $A = (x_i, x_j, a)$, $B = (y_i, x_j, a)$, $C = (y_i, y_j, a)$, $D = (x_i, y_j, a)$ (图 17.1.1).

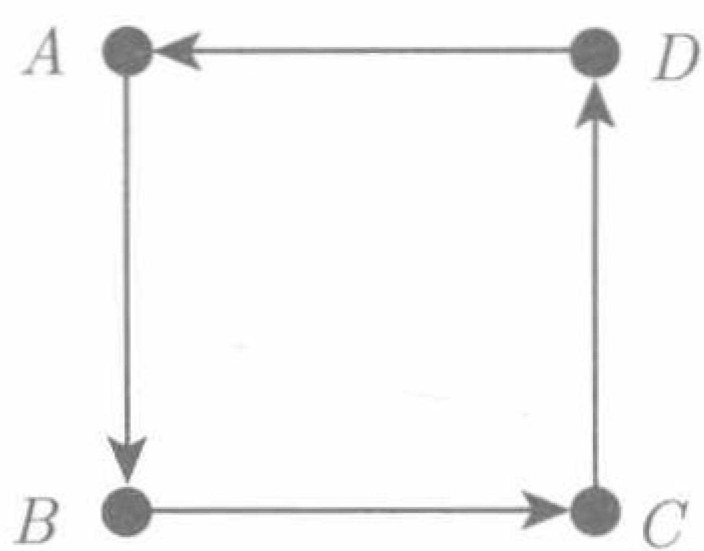


图 17.1.1 闭回路

17.2 势 方 程

本节推导 (加权) 势博弈所满足的基本方程, 称为势方程.

引理 17.2.1 一个有限博弈 G 是加权势博弈, 当且仅当存在 (i) $P_w(x_1, \cdots, x_n)$; (ii) $d_i(x_1, \cdots, \hat{x}_i, \cdots, x_n)$, $i = 1, \cdots, n$, 这里 $\hat{}$ 表示没有该项 (即, d_i 与 x_i 无关); (iii) $w_i > 0$, $i = 1, \cdots, n$, 使得

$$c_i(x_1, \cdots, x_n) = w_i P_w(x_1, \cdots, x_n) + d_i(x_1, \cdots, \hat{x}_i, \cdots, x_n), \quad i = 1, \cdots, n. \tag{17.2.1}$$

这里 P_w 为加权势函数.

将 (17.2.1) 表示为向量形式则得

$$V_i^c \ltimes_{j=1}^n x_j = w_i V_P^w \ltimes_{j=1}^n x_j + V_i^d \ltimes_{j \neq i} x_j, \quad i = 1, \cdots, n, \tag{17.2.2}$$

这里 $V_i^c, V_P^w \in \mathbb{R}^k$ 以及 $V_i^d \in \mathbb{R}^{k/k_i}$ 都是行向量, 是相应函数的结构向量.

因此, 检验 G 是否是势博弈就等价于 (17.2.1) 是否存在相应的 P_w 和 d_i. 这等价于 (17.2.2) 是否存在解 V_P^w 和 V_i^d.

定义

$$E_i := I_{\alpha_i} \otimes \mathbf{1}_{k_i} \otimes I_{\beta_i}, \quad i = 1, \cdots, n, \tag{17.2.3}$$

这里

$$\alpha_1 = 1, \quad \alpha_i = \prod_{j=1}^{i-1} k_j, \quad i \geqslant 2,$$
$$\beta_n = 1, \quad \beta_i = \prod_{j=i+1}^{n} k_j, \quad i \leqslant n-1.$$

那么 (17.2.2) 就可以表示成

$$V_i^d E_i^{\mathrm{T}} = V_i^c - w_i V_P^w, \quad i = 1, \cdots, n. \tag{17.2.4}$$

从 (17.2.4) 解出

$$w_1 V_P^w = V_1^c - V_1^d E_1^{\mathrm{T}}.$$

代入 (17.2.4) 的其他方程可得

$$w_1 V_i^d E_i^{\mathrm{T}} - w_i V_1^d E_1^{\mathrm{T}} = w_1 V_i^c - w_i V_1^c, \quad i = 2, 3, \cdots, n. \tag{17.2.5}$$

定义两组向量如下:

$$\begin{aligned} \xi_i &:= (V_i^d)^{\mathrm{T}}, & i &= 1, \cdots, n, \\ b_{i-1} &:= [w_1 V_i^c - w_i V_1^c]^{\mathrm{T}}, & i &= 2, \cdots, n. \end{aligned} \tag{17.2.6}$$

那么, (17.2.2) 可表达为

$$E^w \xi = b, \tag{17.2.7}$$

这里

$$\xi = (\xi_1^{\mathrm{T}}, \cdots, \xi_n^{\mathrm{T}})^{\mathrm{T}},$$
$$b = (b_1^{\mathrm{T}}, \cdots, b_{n-1}^{\mathrm{T}})^{\mathrm{T}},$$

且

$$E^w = \begin{bmatrix} -w_2 E_1 & w_1 E_2 & 0 & \cdots & 0 \\ -w_3 E_1 & 0 & w_1 E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ -w_n E_1 & 0 & 0 & \cdots & w_1 E_n \end{bmatrix}. \tag{17.2.8}$$

综合以上的讨论可知:

定理 17.2.1 设 $G = (N, S, C)$ 为一有限博弈, $|N| = n$, $|S_i| = k_i$, $i = 1, \cdots, n$. G 是一个以 $\{w_i > 0 \mid i = 1, \cdots, n\}$ 为权的加权势博弈, 当且仅当方程 (17.2.7) 有解. 并且, 如果解存在, 则

$$V_P^w = \frac{1}{w_1}\left[V_1^c - \xi_1^{\mathrm{T}} E_1^{\mathrm{T}}\right]. \tag{17.2.9}$$

注 (i) 称 (17.2.7) 为势方程. 当 $w_i=1,\ i=1,\cdots,n$, 加权势博弈变为势博弈. 记其系数方程为

$$E := E^w\big|_{w=\mathbf{1}_n^{\mathrm{T}}}.$$

(ii) 如果 P^w 和 $\tilde{P}^w$ 是权重为 w 的两个 (加权) 势函数. 定义一个新博弈 G', 它将原支付函数改为

$$c_i' = \frac{c_i}{w_i}, \quad i=1,\cdots,n.$$

那么, P^w 和 $\tilde{P}^w$ 都是 G' 的势函数. 根据定理 17.1.1,

$$\tilde{P}^w - P^w = c_0 \in \mathbb{R}.$$

(iii) 由 (ii) 可知

$$\operatorname{rank}(E^w) = \sum_{i=1}^n \left(\frac{k}{k_i}\right) - 1 := r_E,$$

它比 E^w 的列数小 1.

(iv) 记

$$k_{-i} = \frac{k}{k_i}, \quad i=1,\cdots,n.$$

那么, 容易检验

$$\xi_0 := \left[w_1\mathbf{1}_{k_{-1}}^{\mathrm{T}}, w_2\mathbf{1}_{k_{-2}}^{\mathrm{T}}, \cdots, w_n\mathbf{1}_{k_{-n}}^{\mathrm{T}}\right]^{\mathrm{T}}$$

是 (17.2.7) 相应的齐次方程的解, 即

$$E^w\xi_0 = 0.$$

(v) 由 (iii) 及 (iv) 可推知, E^w 的任何 r_E 列均为 $\operatorname{span}\{\operatorname{Col}(E^w)\}$ 的一组基底.

17.3 势博弈的验证

本节通过几个例子说明如何应用势方程检验势博弈.

例 17.3.1 一个有限博弈 G, $|N|=3$, $|S_i|=2$, $i=1,2,3$, 支付矩阵见表 17.3.1.

表 17.3.1 例 17.3.1 支付矩阵

支付 \ 局势	111	112	121	122	211	212	221	222
c_1	a	b	b	d	c	e	e	f
c_2	a	b	c	e	b	d	e	f
c_3	a	c	b	e	b	e	d	f

我们检验 G 是否为势博弈?

利用式 (17.2.3) 可得

$$
\begin{aligned}
E_1 &= (\delta_2[1,2,1,2])^{\mathrm{T}} \otimes I_2 \\
&= \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}}; \qquad (17.3.1)
\end{aligned}
$$

$$
\begin{aligned}
E_2 &= (\delta_2[1,1,2,2])^{\mathrm{T}} \otimes I_2 \\
&= \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}^{\mathrm{T}}; \qquad (17.3.2)
\end{aligned}
$$

$$
\begin{aligned}
E_3 &= (\delta_4[1,1,2,2,3,3,4,4])^{\mathrm{T}} \\
&= \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}^{\mathrm{T}}. \qquad (17.3.3)
\end{aligned}
$$

因此可得

$$
E = \begin{bmatrix}
-1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}.
$$

下面计算

$$b_1 = V_2^c - V_1^c = [0,0,c-b,e-d,b-c,d-e,0,0]^{\mathrm{T}},$$
$$b_2 = V_3^c - V_1^c = [0,c-b,0,e-d,b-c,0,d-e,0]^{\mathrm{T}}.$$

于是

$$\begin{aligned} b &= \left[b_1^{\mathrm{T}}\ b_2^{\mathrm{T}}\right]^{\mathrm{T}} \\ &= [0,0,\alpha,\beta,-\alpha,-\beta,0,0,0,\alpha,0,\beta,-\alpha,0,-\beta,0]^{\mathrm{T}}, \end{aligned}$$

这里 $\alpha = c-b$, $\beta = e-d$. 不难检验

$$\begin{aligned} b =& (\alpha+\beta)\mathrm{Col}_1(E) + \beta\mathrm{Col}_2(E) + \beta\mathrm{Col}_3(E) \\ &+ (\alpha+\beta)\mathrm{Col}_5(E) + \beta\mathrm{Col}_6(E) + \beta\mathrm{Col}_7(E) \\ &+ (\alpha+\beta)\mathrm{Col}_9(E) + \beta\mathrm{Col}_{10}(E) + \beta\mathrm{Col}_{11}(E). \end{aligned}$$

根据定理 17.2.1 可知, 当 $|N| = 3$ $|S_i| = 2$, $i = 1,2,3$ 时, 对称博弈均为势博弈 (这里, 对称博弈指支付对玩家对称).

下面, 为计算势函数, 给出具体参数. 设 $a = 1$, $b = 1$, $c = 2$, $d = -1$, $e = 1$, $f = -1$. 则不难算出

$$b_1 = [V_2^c - V_1^c]^{\mathrm{T}} = [0,0,1,2,-1,-2,0,0]^{\mathrm{T}},$$
$$b_2 = [V_3^c - V_1^c]^{\mathrm{T}} = [0,1,0,2,-1,0,-2,0]^{\mathrm{T}}.$$

解势方程 (17.2.7), 任求一个解 $\xi = [3,2,2,0,3,2,2,0,3,2,2,0]^{\mathrm{T}}$. 则 $V_1^d = \xi_1^{\mathrm{T}} = [3,2,2,0]$. 利用 (17.2.9) 可得

$$\begin{aligned} V_P &= V_1^c - V_1^d D_r[2,2] \\ &= [1,1,1,-1,2,1,1,-1] - [3,2,2,0]\delta_2[1,2,1,2] \\ &= [-2,-1,-1,-1,-1,-1,-1,-1]. \end{aligned}$$

(这里和后面用到的 $D_r[a,b]$, $D_f[a,b]$ 的定义见命题 14.1.1.)

最后可得势函数

$$P(x) = [-2,-1,-1,-1,-1,-1,-1,-1]x + c_0,$$

这里 $x = \ltimes_{i=1}^3 x_i \in \Delta_8$.

下面考察另一个例子: 石头–剪刀–布.

例 17.3.2 考虑两人玩石头–剪刀–布, 支付见表 17.3.2, 表中: 1: 石头, 2: 剪刀, 3: 布.

表 17.3.2 石头–剪刀–布的支付矩阵

支付 \ 局势	11	12	13	21	22	23	31	32	33
c_1	0	−1	1	1	0	−1	−1	1	0
c_2	0	1	−1	−1	0	1	1	−1	0

G 是势博弈吗?

容易算得

$$E_1 = \delta_3[1,2,3,1,2,3,1,2,3]^{\mathrm{T}}$$
$$E_2 = \delta_3[1,1,1,2,2,2,3,3,3]^{\mathrm{T}}.$$

于是有

$$E = [-E_1\ E_2] = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix},$$

$$b = V_2^c - V_1^c = [0,2,-2,-2,0,2,2,-2,0]^{\mathrm{T}}.$$

不难检验: $\mathrm{rank}(E) = 5$ 而 $\mathrm{rank}[E\ b] = 6$, 势方程无解. 因此, 石头–剪刀–布不是一个势博弈.

最后注意一点. 势函数与李雅普诺夫函数一样, 是用来控制收敛的. 其实, 势函数在某种意义上就是一个李雅普诺夫函数.

命题 17.3.1　*设一演化博弈是以串联或级联短视最优响应 (MBRA) 更新的泛势博弈. 那么, 其势函数也是一个李雅普诺夫函数.*

17.4　网络演化博弈的势

先考虑一个例子.

例 17.4.1　考察一个网络演化博弈 $((N,E),G,\Pi)$, 设网络图如图 17.4.1 或图 17.4.2 所示.

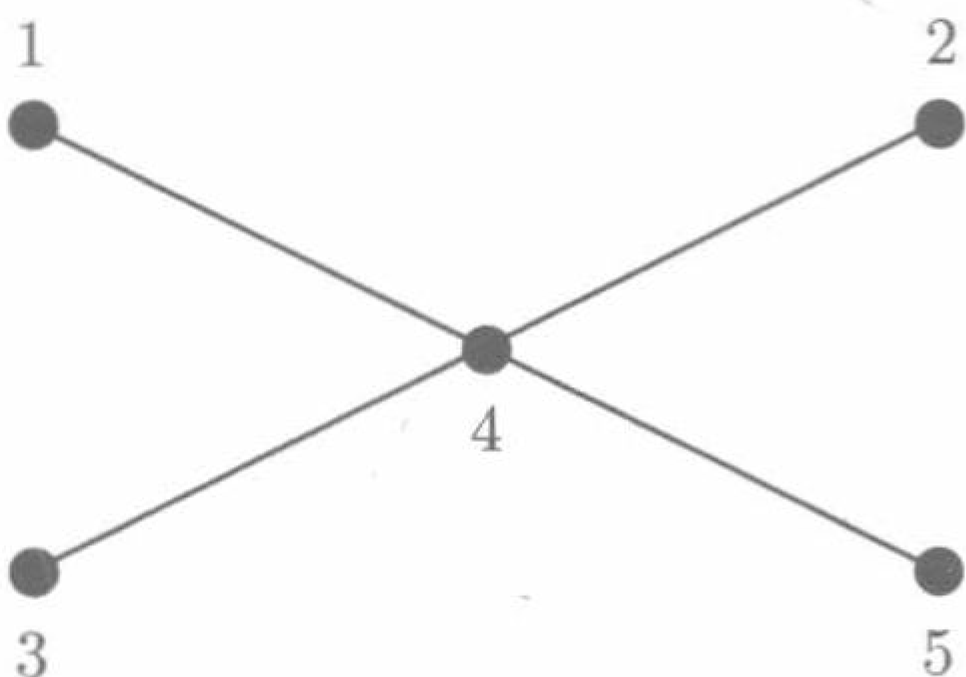


图 17.4.1　网络图 (a)

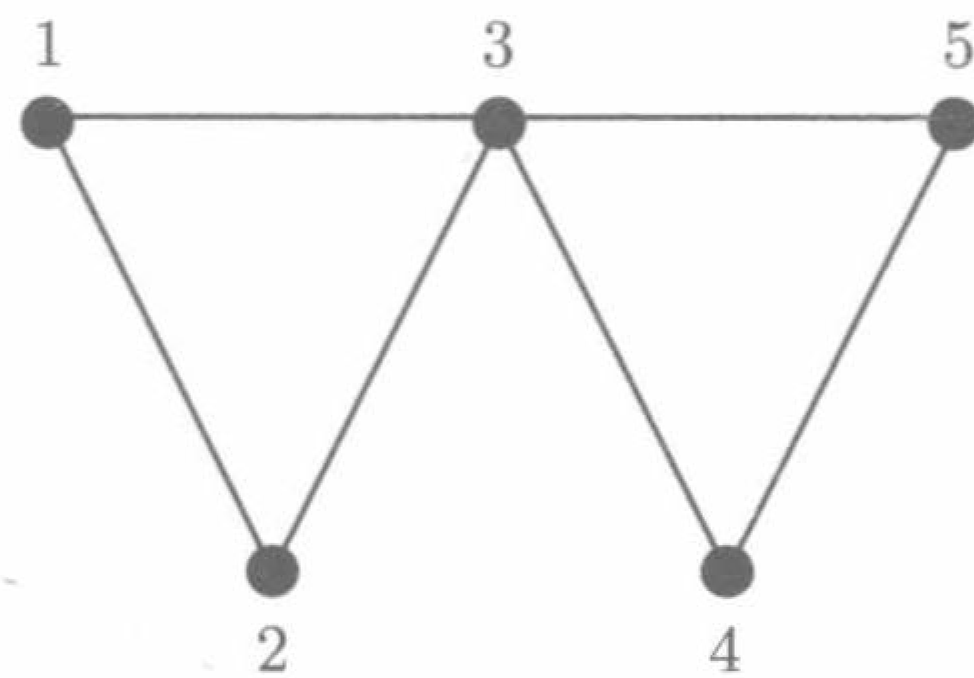


图 17.4.2 网络图 (b)

设基本网络博弈为囚徒困境, 支付矩阵见表 16.5.1, 其中 $R=-1$, $S=-10$, $T=0$, $P=-5$. 由于势矩阵 E 只依赖于 $|N|$ 和 $|S_i|=k, \forall i$, 与网络图无关. 于是可得到 E 如下

$$E=\begin{bmatrix} -1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 \end{bmatrix} \in \mathcal{M}_{128\times 80}.$$

(为节约空间, 只写出几个元素.) 下面计算 V_i^c, $i=1,\cdots,5$.

1) (网络图 (a)): 不难验证

$$\begin{aligned} V_1^c = & [-1 \quad -1 \quad -10 \quad -10 \quad -1 \quad -1 \quad -10 \quad -10 \\ & -1 \quad -1 \quad -10 \quad -10 \quad -1 \quad -1 \quad -10 \quad -10 \\ & 0 \quad 0 \quad -5 \quad -5 \quad 0 \quad 0 \quad -5 \quad -5 \\ & 0 \quad 0 \quad -5 \quad -5 \quad 0 \quad 0 \quad -5 \quad 5]. \end{aligned}$$

$$\begin{aligned} V_2^c = & [-1 \quad -1 \quad -10 \quad -10 \quad -1 \quad -1 \quad -10 \quad -10 \\ & 0 \quad 0 \quad -5 \quad -5 \quad 0 \quad 0 \quad -5 \quad -5 \\ & -1 \quad -1 \quad -10 \quad -10 \quad -1 \quad -1 \quad -10 \quad -10 \\ & 0 \quad 0 \quad -5 \quad -5 \quad 0 \quad 0 \quad -5 \quad -5]. \end{aligned}$$

$$\begin{aligned} V_3^c = & [-1 \quad -1 \quad -10 \quad -10 \quad 0 \quad 0 \quad -5 \quad -5 \\ & -1 \quad -1 \quad -10 \quad -10 \quad 0 \quad 0 \quad -5 \quad -5 \\ & -1 \quad -1 \quad -10 \quad -10 \quad 0 \quad 0 \quad -5 \quad -5 \\ & -1 \quad -1 \quad -10 \quad -10 \quad 0 \quad 0 \quad -5 \quad -5]. \end{aligned}$$

$$\begin{aligned} V_4^c = & [-4 \quad -13 \quad 0 \quad -5 \quad -13 \quad -22 \quad -5 \quad -10 \\ & -13 \quad -22 \quad -5 \quad -10 \quad -22 \quad -31 \quad -10 \quad -15 \\ & -13 \quad -22 \quad -5 \quad -10 \quad -22 \quad -31 \quad -10 \quad -15 \\ & -22 \quad -31 \quad -10 \quad -15 \quad -31 \quad -40 \quad -15 \quad -20]. \end{aligned}$$

$$\begin{aligned} V_5^c = [&-1 \quad 0 \quad -10 \quad -5 \quad -1 \quad 0 \quad -10 \quad -5 \\ &-1 \quad 0 \quad -10 \quad -5 \quad -1 \quad 0 \quad -10 \quad -5 \\ &-1 \quad 0 \quad -10 \quad -5 \quad -1 \quad 0 \quad -10 \quad -5 \\ &-1 \quad 0 \quad -10 \quad -5 \quad -1 \quad 0 \quad -10 \quad -5]. \end{aligned}$$

根据定理 17.2.1 不难检验, 该网络演化博弈为势博弈. 同时, 由 (17.2.7) 的一组解可得

$$\begin{aligned} \xi_1 = [&28 \quad 27 \quad 15 \quad 10 \quad 27 \quad 26 \quad 10 \quad 5 \\ &27 \quad 26 \quad 10 \quad 5 \quad 26 \quad 25 \quad 5 \quad 0]. \end{aligned}$$

利用 (17.2.9), 可算出势函数:

$$\begin{aligned} V_P^a = [&-29 \quad -28 \quad -25 \quad -20 \quad -28 \quad -27 \quad -20 \quad -15 \\ &-28 \quad -27 \quad -20 \quad -15 \quad -27 \quad -26 \quad -15 \quad -10 \\ &-28 \quad -27 \quad -20 \quad -15 \quad -27 \quad -26 \quad -15 \quad -10 \\ &-27 \quad -26 \quad -15 \quad -10 \quad -26 \quad -25 \quad -10 \quad -5]. \end{aligned}$$

2) (网络图 (b)): 虽然网络图改变了, 但仍可证明, 这个网络演化博弈也是势博弈, 势函数为

$$\begin{aligned} V_P^b = [&-46 \quad -44 \quad -44 \quad -38 \quad -42 \quad -36 \quad -36 \quad -26 \\ &-44 \quad -42 \quad -42 \quad -36 \quad -36 \quad -30 \quad -30 \quad -20 \\ &-44 \quad -42 \quad -42 \quad -36 \quad -36 \quad -30 \quad -30 \quad -20 \\ &-38 \quad -36 \quad -36 \quad -30 \quad -26 \quad -20 \quad -20 \quad -10]. \end{aligned}$$

不难验证, 在上例中不管如何改变网络图, 得到的网络演化博弈都是势博弈. 实际上, 我们可以证明如下定理:

定理 17.4.1 ([64]) *给定一个网络演化博弈* $((N,E),G,\Pi)$. *如果它的基本网络博弈是势博弈, 则整个网络演化博弈也是势博弈. 而且, 整个网络的势函数是所有博弈的势函数之和. 即*

$$P = \sum_{(i,j)\in E} P^{(i,j)}, \tag{17.4.1}$$

这里 $P^{(i,j)}$ *是发生在* i, j *之间的基本网络博弈的势函数.*

例 17.4.2 重新考察例 17.4.1. 容易算出囚徒困境的势函数结构向量为

$$V_0 = (R-T, 0, 0, P-S).$$

也就是说, 对任意 $(i,j)\in E$ 有势函数

$$P^{(i,j)}(x_i, x_j) = V_0 x_i x_j, \tag{17.4.2}$$

这里

$$V_0 = (R - T, 0, 0, P - S) = (-1\ 0\ 0\ 5).$$

为求和, 将 (17.4.2) 变为一般形式

$$P^{(i,j)}(x) = P^{(i,j)}(x_i, x_j) = V_0 x_i x_j := V_P^{(i,j)} x, \tag{17.4.3}$$

这里 $x = \ltimes_{i=1}^5 x_i$. 由于

$$P^{(1,2)}(x_1, x_2) = V_0 x_1 x_2 = V_0 D_r[4,8] x_1 x_2 x_3 x_4 x_5, \tag{17.4.4}$$

可得

$$\begin{aligned} V_P^{(1,2)} &= V_0 D_r[4,8] = V_0 \left(I_4 \otimes \mathbf{1}_8^{\mathrm{T}}\right) \\ &= [\begin{matrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \end{matrix}]. \end{aligned}$$

类似地, 所有 $V_P^{(i,j)}$, $(i,j) \in E$ 可计算如下:

$$\begin{aligned} &V_P^{(1,3)} = V_0 D_r[2,2] D_r[8,2], && V_P^{(1,4)} = V_0 D_r[2,4] D_r[16,2], \\ &V_P^{(1,5)} = V_0 D_r[2,8], && V_P^{(2,3)} = V_0 D_f[2,2] D_r[8,4], \\ &V_P^{(2,4)} = V_0 D_f[2,2] D_r[4,2] D_r[16,2], && V_P^{(2,5)} = V_0 D_f[2,2] D_r[4,4], \\ &V_P^{(3,4)} = V_0 D_f[4,2] D_r[16,2], && V_P^{(3,5)} = V_0 D_f[4,2] D_r[8,2], \\ &V_P^{(4,5)} = V_0 D_f[8,2]. \end{aligned}$$

下面, 用 (17.4.1) 重新计算例 17.4.1 中网络演化博弈的势函数. 为区别, 将它们记为 $\tilde{P}$.

1) 考察网络图 17.4.1. 利用 (17.4.1) 可得

$$\begin{aligned} V_{\tilde{P}}^a &= V_P^{(1,4)} + V_P^{(2,4)} + V_P^{(3,4)} + V_P^{(4,5)} \\ &= [\begin{matrix} -4 & -3 & 0 & 5 & -3 & -2 & 5 & 10 \\ -3 & -2 & 5 & 10 & -2 & -1 & 10 & 15 \\ -3 & -2 & 5 & 10 & -2 & -1 & 10 & 15 \\ -2 & -1 & 10 & 15 & -1 & 0 & 15 & 20 \end{matrix}]. \end{aligned}$$

与例 17.4.1 中得到的 V_P^a 相比, 注意到

$$P^a(x) := V_P^a x; \quad \tilde{P}^a(x) = V_{\tilde{P}}^a x,$$

于是有

$$\tilde{P}^a(x) = P^a(x) + 25.$$

即除一常数外, 它们相等.

2) 考察网络图 17.4.2. 有

$$\begin{aligned} V_{\tilde{P}}^b &= V_P^{(1,2)} + V_P^{(1,3)} + V_P^{(2,3)} + V_P^{(3,4)} + V_P^{(3,5)} + V_P^{(4,5)} \\ &= [-6 \quad -4 \quad -4 \quad 2 \quad -2 \quad 4 \quad 4 \quad 14 \\ &\quad\ -4 \quad -2 \quad -2 \quad 4 \quad 4 \quad 10 \quad 10 \quad 20 \\ &\quad\ -4 \quad -2 \quad -2 \quad 4 \quad 4 \quad 10 \quad 10 \quad 20 \\ &\quad\ 2 \quad 4 \quad 4 \quad 10 \quad 14 \quad 20 \quad 20 \quad 30]. \end{aligned}$$

与例 17.4.1 中得到的 V_P^b 相比, 可知

$$\tilde{P}^b(x) = P^b(x) + 40.$$

17.5 注释与参考

势博弈最早是由 Rosenthal 提出的[126]. Shapley 等做了许多进一步的完善和发展[112]. 势博弈作为一类特殊的博弈有许多优良的性质. 特别是它在演化下会收敛到纳什均衡点的这一特性, 使它倍受青睐. 它也因此在许多实际系统的研究中得到应用: 例如, 多自主体控制[14, 80]、图形覆盖[149, 156]、交通拥塞问题[109, 144]、电力配给与电网控制[25, 85] 等. 本节内容主要基于文献 [64].

第 18 章　有限博弈的空间分解

18.1　有限博弈的向量空间结构

一个有限博弈可以表示为 $G=(N,S,C)$, 这里 $N=\{1,2,\cdots,n\}$ 为玩家; $S=\prod_{i=1}^{n} S_i$ 称为局势, 其因子 $S_i=\{1,2,\cdots,k_i\}$ 为第 i 个玩家的策略集, 当 n 和 k_i, $i=1,\cdots,n$ 都确定之后, 所有这样的博弈记为 $\mathcal{G}_{[n;k_1,\cdots,k_n]}$. 注意到所有 $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 中的博弈, 它们的区别仅在于支付函数的不同. 支付函数 $c_i: S\to\mathbb{R}$ 是一个伪逻辑函数, 因此, 在向量形式下可表示为

$$c_i(x_1,\cdots,x_n)=V_i^c \ltimes_{j=1}^{n} x_j,\quad i=1,\cdots,n, \tag{18.1.1}$$

这里 $V_i^c\in\mathbb{R}^k$ 是一个行向量, 称为 c_i 的结构向量 $\left(k=\prod_{i=1}^{n} k_i\right)$. 记

$$V^c=(V_1^c,V_2^c,\cdots,V_n^c)\in\mathbb{R}^{nk},$$

它称为 G 的结构向量. 于是可知, 每一个博弈 $G\in\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 对应于唯一的一个 $V_G^c\in\mathbb{R}^{nk}$. 反之亦然. 因此, $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 具有一个自然的向量空间结构, 它与 $\mathbb{R}^{nk}$ 同构,

$$\mathcal{G}_{[n;k_1,\cdots,k_n]}\cong\mathbb{R}^{nk}. \tag{18.1.2}$$

注意到对每一个有限博弈 $G\in\mathcal{G}_{[n;k_1,\cdots,k_n]}$, 其结构向量 V_G^c 完全决定了 G 的性质, 因此, (18.1.2) 给出的向量空间结构是十分自然而合理的. $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的向量空间结构由文献 [34] 最早提出. 文献 [34] 进而研究了这个空间的结构分解, 提出将其分解为 3 个子空间的直和. 这 3 个子空间分别为: (i) 纯势子空间$\mathcal{P}$; (ii) 非策略子空间$\mathcal{N}$; 和 (iii) 纯调和子空间$\mathcal{H}$, 使得 $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 可以分解成 $\mathcal{P}$, $\mathcal{N}$, 和 $\mathcal{H}$ 的直和:

$$\mathcal{G}_{[n;k_1,\cdots,k_n]}=\underbrace{\mathcal{P}\uplus\overbrace{\mathcal{N}}^{}}_{\text{势博弈}}\overbrace{\uplus\,\mathcal{H}}^{\text{调和博弈}}. \tag{18.1.3}$$

在直交分解(18.1.3) 中还给出了另外两个子空间, (i) 势博弈子空间, 它是纯势子空间与非策略子空间的直和:

$$\mathcal{G}_P=\mathcal{P}\uplus\mathcal{N};$$

(ii) 调和博弈子空间, 它是纯调和子空间与非策略子空间的直和:

$$\mathcal{G}_H = \mathcal{H} \uplus \mathcal{N}.$$

本章应用代数状态空间方法讨论上述分解.

18.2　势博弈子空间

首先回忆上一章内容, 令权重为 1, 那么, (17.2.7)–(17.2.8) 变为

$$\begin{bmatrix} -E_1 & E_2 & 0 & \cdots & 0 \\ -E_1 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ -E_1 & 0 & 0 & \cdots & E_n \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{bmatrix} = \begin{bmatrix} (V_2^c - V_1^c)^{\mathrm{T}} \\ (V_3^c - V_1^c)^{\mathrm{T}} \\ \vdots \\ (V_n^c - V_1^c)^{\mathrm{T}} \end{bmatrix}, \tag{18.2.1}$$

这里 $\xi_i \in \mathbb{R}^{k/k_i}$. 记

$$E := \begin{bmatrix} -E_1 & E_2 & 0 & \cdots & 0 \\ -E_1 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ -E_1 & 0 & 0 & \cdots & E_n \end{bmatrix}. \tag{18.2.2}$$

定理 17.2.1 告诉我们, $G \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$ 为势博弈, 当且仅当, (17.2.7)–(17.2.8) 有解. 这等价于

$$\begin{bmatrix} (V_2^c - V_1^c)^{\mathrm{T}} \\ (V_3^c - V_1^c)^{\mathrm{T}} \\ \vdots \\ (V_n^c - V_1^c)^{\mathrm{T}} \end{bmatrix} \in \operatorname{span}(E). \tag{18.2.3}$$

注意到在 (18.2.3) 中可以自由地选择 V_1^c, 因此, (18.2.3) 可改写成

$$\begin{bmatrix} (V_1^c)^{\mathrm{T}} \\ (V_2^c - V_1^c)^{\mathrm{T}} \\ (V_3^c - V_1^c)^{\mathrm{T}} \\ \vdots \\ (V_n^c - V_1^c)^{\mathrm{T}} \end{bmatrix} \in \operatorname{span}(E^e), \tag{18.2.4}$$

这里

$$E^e = \begin{bmatrix} I_k & 0 \\ 0 & E \end{bmatrix}.$$

进一步等价地将其改写为

$$\begin{bmatrix} I_k & 0 & \cdots & 0 \\ -I_k & I_k & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ -I_k & 0 & \cdots & I_k \end{bmatrix} \begin{bmatrix} (V_1^c)^{\mathrm{T}} \\ (V_2^c)^{\mathrm{T}} \\ (V_3^c)^{\mathrm{T}} \\ \vdots \\ (V_n^c)^{\mathrm{T}} \end{bmatrix} \in \operatorname{span}(E^e). \tag{18.2.5}$$

于是有

$$(V_G^c)^{\mathrm{T}} \in \operatorname{span}(E_P), \tag{18.2.6}$$

这里

$$\begin{aligned} E_P &:= \begin{bmatrix} I_k & 0 & \cdots & 0 \\ -I_k & I_k & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ -I_k & 0 & \cdots & I_k \end{bmatrix}^{-1} E^e \\ &= \begin{bmatrix} I_k & 0 & 0 & 0 & \cdots & 0 \\ I_k & -E_1 & E_2 & 0 & \cdots & 0 \\ I_k & -E_1 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ I_k & -E_1 & 0 & 0 & \cdots & E_n \end{bmatrix}. \end{aligned} \tag{18.2.7}$$

由上一章可知, 删去 E 的任意一列, 余下的各列变为 $\operatorname{span}(E)$ 的一个基底. 比较 E_P 与 E, 我们不妨删去 E_n 的最后一列, 将 E_n 删余部分记为 E_n^0, 然后定义

$$E_P^0 := \begin{bmatrix} I_k & 0 & 0 & 0 & \cdots & 0 \\ I_k & -E_1 & E_2 & 0 & \cdots & 0 \\ I_k & -E_1 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ I_k & -E_1 & 0 & 0 & \cdots & E_n^0 \end{bmatrix}.$$

于是, 不难看出

$$\operatorname{span}(E_P) = \operatorname{span}(E_P^0).$$

而且, 因为 E_P^0 各列线性无关, 它是 $\text{span}(E_P)$ 的一组基.

根据以上讨论可得如下结果:

定理 18.2.1 势博弈子空间为

$$\mathcal{G}_P = \text{span}(E_P), \tag{18.2.8}$$

它的一个基底为 $\text{Col}(E_P^0)$.

由 E_P^0 的结构不难发现

推论 18.2.1 1) $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的势博弈子空间 $\mathcal{G}_P$ 的维数为

$$\dim(\mathcal{G}_P) = k + \sum_{j=1}^{n} \frac{k}{k_j} - 1. \tag{18.2.9}$$

2) 纯调和子空间 $\mathcal{H}$ 的维数为

$$\dim(\mathcal{H}) = (n-1)k - \sum_{j=1}^{n} \frac{k}{k_j} + 1. \tag{18.2.10}$$

18.3 非策略子空间

定义 18.3.1 考察两个博弈 G, $\tilde{G} \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$. G 和 $\tilde{G}$ 称为策略等价的, 如果对任意 $i \in N$, 两个策略 x_i, $y_i \in S_i$, 以及任意的 $x^{-i} \in S^{-i}$ (这里 $S^{-i} = \prod\limits_{j \neq i} S_j$), 都有

$$c_i(x_i, x^{-i}) - c_i(y_i, x^{-i}) = \tilde{c}_i(x_i, x^{-i}) - \tilde{c}_i(y_i, x^{-i}). \tag{18.3.1}$$

策略等价的物理意义是, 在两个博弈中, 对应玩家策略改变所引起的效果是相同的. 也就是说, 相应策略在博弈中所起的作用是相同的.

引理 18.3.1 两个博弈 G, $\tilde{G} \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$ 是策略等价的, 当且仅当, 对每一个 $x^{-i} \in S^{-i}$ 都存在一个 $d_i(x^{-i})$ 使得

$$\begin{aligned} & c_i(x_i, x^{-i}) - \tilde{c}_i(x_i, x^{-i}) = d_i(x^{-i}), \\ & \forall x_i \in S_i,\ \forall x^{-i} \in S^{-i}, i = 1, \cdots, n. \end{aligned} \tag{18.3.2}$$

证明 (必要性) 利用反证, 设 (18.3.2) 不成立. 则存在某个 i 以及一个 $x^{-i} \in S^{-i}$, 使得 $c_i(x_i, x^{-i}) - \tilde{c}_i(x_i, x^{-i})$ 依赖于 x_i. 即, 存在 $a_i, b_i \in S_i$ 使得

$$c_i(a_i, x^{-i}) - \tilde{c}_i(a_i, x^{-i}) \neq c_i(b_i, x^{-i}) - \tilde{c}_i(b_i, x^{-i}).$$

于是

$$c_i(a_i, x^{-i}) - c_i(b_i, x^{-i}) \neq \tilde{c}_i(a_i, x^{-i}) - \tilde{c}_i(b_i, x^{-i}),$$

这与 (18.3.2) 矛盾.

(充分性) 由 (18.3.2) 有

$$c_i(x_i, x^{-i}) = \tilde{c}_i(x_i, x^{-i}) + d_i(x^{-i}), \quad \forall x_i \in S_i.$$

代入 (18.3.1) 则得所需等式. □

为方便计, 定义一个记号

$$k^{[\alpha,\beta]} := \begin{cases} \prod_{j=\alpha}^{\beta} k_j, & \alpha \leqslant \beta, \\ 1, & \alpha > \beta. \end{cases}$$

分别记 c_i, $\tilde{c}_i$, 以及 d_i 的结构向量为 V_i^c, $\tilde{V}_i^c$, 以及 V_i^d, 则可将 (18.3.2) 表示为矩阵形式如下:

$$\begin{aligned} &V_i^c \ltimes_{j=1}^n x_j - \tilde{V}_i^c \ltimes_{j=1}^n x_j = V_i^d \ltimes_{j\neq i}^n x_j \\ &= V_i^d \left(I_{k^{[1,i-1]}} \otimes \mathbf{1}_{k_i}^{\mathrm{T}} \otimes I_{k^{[i+1,n]}} \right) \ltimes_{j=1}^n x_j. \end{aligned}$$

因而可得

$$B_N^i (V_i^d)^{\mathrm{T}} = (V_i^c - \tilde{V}_i^c)^{\mathrm{T}}, \tag{18.3.3}$$

这里

$$\begin{aligned} B_N^i &:= I_{k^{[1,i-1]}} \otimes \mathbf{1}_{k_i} \otimes I_{k^{[i+1,n]}} \\ &= E_i, \quad i = 1, \cdots, n. \end{aligned} \tag{18.3.4}$$

综上可得

定理 18.3.1 G 与 $\tilde{G}$ 策略等价, 当且仅当,

$$\left(V^c - \tilde{V}^c \right)^{\mathrm{T}} \in \operatorname{span}(B_N), \tag{18.3.5}$$

这里

$$B_N = \begin{bmatrix} E_1 & 0 & \cdots & 0 \\ 0 & E_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & E_n \end{bmatrix}. \tag{18.3.6}$$

定义 18.3.2 子空间

$$\mathcal{N} := \operatorname{span}(B_N)$$

称为非策略子空间.

由定理 18.3.1 可知: G 与 $\tilde{G}$ 策略等价, 当且仅当, 存在一个向量 $\eta \in \mathcal{N}$, 使得

$$\left(\tilde{V}^c\right)^{\mathrm{T}} = (V^c)^{\mathrm{T}} + \eta. \tag{18.3.7}$$

注意到 E_i 有 k/k_i 列, 且它们线性无关, $i = 1, \cdots, n$, 于是有

推论 18.3.1　1) 子空间 $\mathcal{N}$ 的维数为

$$\dim(\mathcal{N}) = \sum_{i=1}^{n} \frac{k}{k_i}. \tag{18.3.8}$$

2) 子空间 $\mathcal{P}$ 的维数为

$$\dim(\mathcal{P}) = k - 1. \tag{18.3.9}$$

定义

$$B_P := \begin{bmatrix} I_k & E_1 & 0 & 0 & \cdots & 0 \\ I_k & 0 & E_2 & 0 & \cdots & 0 \\ I_k & 0 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ I_k & 0 & 0 & 0 & \cdots & E_n \end{bmatrix}. \tag{18.3.10}$$

比较 (18.3.10) 与 (18.2.6), 立可发现

$$\mathcal{G}_P = \mathrm{span}(B_P) = \mathrm{span}(E_P).$$

删去 B_P 的最后一列 (或者说, 将 B_P 中的 E_n 换为 E_n^0), 余下的矩阵记作

$$B_P^0 := \begin{bmatrix} I_k & E_1 & 0 & 0 & \cdots & 0 \\ I_k & 0 & E_2 & 0 & \cdots & 0 \\ I_k & 0 & 0 & E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ I_k & 0 & 0 & 0 & \cdots & E_n^0 \end{bmatrix}. \tag{18.3.11}$$

则显然 $\mathrm{Col}\left(B_P^0\right)$ 是 $\mathcal{G}_P$ 的一个基底.

重新观察 (18.3.10), 立可看出

推论 18.3.2　子空间 $\mathcal{N}$ 也是 $\mathcal{G}_P$ 的线性子空间. 即

$$\mathcal{N} \subset \mathcal{G}_P.$$

一个有限博弈 $G \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$, 如果其结构向量 $(V^c)^{\mathrm{T}} \in \mathcal{N}$, 则 G 称为一个非策略博弈. 根据 $B_{\mathcal{N}}$ 的构造, 下面的结论是显然的:

定理 18.3.2 如果 G 是一个非策略博弈, 那么, 对任何 $s^{-i} \in S^{-i}$ 均有

$$c_i(x_i, s^{-i}) = \text{const.}, \quad \forall x_i \in S_i. \tag{18.3.12}$$

例 18.3.1 给定一个有限博弈, 设 $N = \{1,2\}$, $S_1 = \{1,2\}$, $S_2 = \{1,2,3\}$, 那么, G 是一个非策略博弈, 当且仅当,

$$(V^c)^{\mathrm{T}} \in \mathcal{N}.$$

等价地,

$$(V_i^c)^{\mathrm{T}} \in \operatorname{span}(E_i), \quad i = 1, 2.$$

由于

$$E_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

于是

$$\begin{aligned} V_1^c &= a[1\ 0\ 0\ 1\ 0\ 0] + b[0\ 1\ 0\ 0\ 1\ 0] + c[0\ 0\ 1\ 0\ 0\ 1] \\ &= [a\ b\ c\ a\ b\ c]. \end{aligned}$$

类似地, 有

$$E_2 = I_2 \otimes \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

于是

$$\begin{aligned} V_2^c &= d[1\ 1\ 1\ 0\ 0\ 0] + e[0\ 0\ 0\ 1\ 1\ 1] \\ &= [d\ d\ d\ e\ e\ e]. \end{aligned}$$

根据它们, 可构造支付双矩阵 18.3.1.

由表 18.3.1 可知

(i) $\dim(\mathcal{N}) = 5$, 它符合公式 (18.3.8);

(ii) 只要 $x_2 \in S_2$ (或 $x_1 \in S_1$) 固定, 则支付函数 P_1 (相应地, P_2) 为常数, 不管乙方的策略如何选择.

表 18.3.1 例 18.3.1 支付双矩阵

$P_1 \backslash P_2$	1	2	3
1	$a,\ d$	$b,\ d$	$c,\ d$
2	$a,\ e$	$b,\ e$	$c,\ e$

18.4 子空间 $\mathcal{P}$ 和 $\mathcal{N}^{\perp}$

首先定义一个辅助空间如下:

$$\mathcal{V} := \operatorname{span}(V),$$

这里

$$V = \begin{bmatrix} I_k \\ I_k \\ \vdots \\ I_k \end{bmatrix} \in \mathcal{M}_{nk \times k}.$$

于是显然有

$$\mathcal{G}_P = \mathcal{V} + \mathcal{N}. \tag{18.4.1}$$

注意到 (18.4.1) 不是一个直和. 实际上, 有如下结论:

命题 18.4.1

$$\mathcal{V} \bigcap \mathcal{N} = \operatorname{span}(\mathbf{1}_{nk}). \tag{18.4.2}$$

证明 首先, 根据 $\mathcal{G}_P$ 和 $\mathcal{N}$ 的维数可知

$$\dim\left(\mathcal{V} \bigcap \mathcal{N}\right) = 1.$$

因此, 要证明 (18.4.1) 只要证

$$\mathbf{1}_{nk} \in \mathcal{V} \bigcap \mathcal{N}. \tag{18.4.3}$$

显见 $\mathbf{1}_{nk} \in \mathcal{V}$, 因此, 只要证明

$$\mathbf{1}_{nk} \in \mathcal{N}. \tag{18.4.4}$$

根据 (18.3.6), 只要证明下式就可以了:

$$\mathbf{1}_k \in \mathrm{span}\,(E_i), \quad i=1,\cdots,n. \tag{18.4.5}$$

这是对的, 因为

$$\begin{aligned}&E_i\mathbf{1}_{k/k_i}\\&=\left(I_{k[1,i-1]}\otimes\mathbf{1}_{k_i}\otimes I_{k[i+1,n]}\right)\left(\mathbf{1}_{k[1,i-1]}\otimes 1\otimes\mathbf{1}_{k[i+1,n]}\right)\\&=\mathbf{1}_{k[1,i-1]}\otimes\mathbf{1}_{k_i}\otimes\mathbf{1}_{k[i+1,n]}\\&=\mathbf{1}_k.\end{aligned}$$

□

记 V^0 为由 V 删去任意一列而得到的矩阵. 定义子空间 $\mathcal{V}^0=\mathrm{span}\left(V^0\right)$.

命题 18.4.2 $\mathcal{V}^0$ 为 $\mathcal{P}$ 的一个子空间. 并且,

$$\mathcal{V}^0 \uplus \mathcal{N} = \mathcal{G}_P. \tag{18.4.6}$$

证明 显然 $\mathbf{1}_{nk}\notin\mathrm{span}\left(V^0\right)$. 因此

$$\mathcal{V}=\mathrm{span}\left(V^0,\mathbf{1}_{nk}\right).$$

根据命题 18.4.1, (18.4.6) 可由 (18.4.1) 得到. □

注意: 删去 V 不同的列可能得到不同的 $\mathcal{V}^0$. 这并不奇怪, 因为直和分解并不是唯一的.

下面考虑 $\mathcal{N}^{\perp}$. 记

$$\begin{aligned}\theta^i_j &:= I_{k[1,i-1]}\otimes\left(\delta^1_{k_i}-\delta^j_{k_i}\right)\otimes I_{k[i+1,n]}\\&\in\mathcal{M}_{k\times(k/k_i)},\quad j=2,\cdots,k_i;\ i=1,\cdots,n.\end{aligned} \tag{18.4.7}$$

由直接计算就可以检验下式:

$$E_i^{\mathrm{T}}\theta^i_j=0,\quad j=2,\cdots,k_i;\ i=1,\cdots,n.$$

因此, 对每个 i 均有

$$\mathrm{span}\left(\theta^i_j\right)\subset[\mathrm{span}\,(E_i)]^{\perp},\quad j=2,\cdots,k_i.$$

注意到

$$\mathrm{span}\left(\theta^i_p\right)\bigcap\mathrm{span}\left(\theta^i_q\right)=\{0\},\quad p\neq q,$$

分析其维数, 不难看出

$$[\mathrm{span}\,(E_i)]^{\perp}=\mathrm{span}\left(\theta^i_2,\theta^i_3,\cdots,\theta^i_{k_i}\right). \tag{18.4.8}$$

定义

$$\Theta_j^i := \begin{bmatrix} \mathbf{0}_{(i-1)k\times(k/k_i)} \\ \theta_j^i \\ \mathbf{0}_{(n-i)k\times(k/k_i)} \end{bmatrix}, \quad j=2,\cdots,k_i.$$

考虑到 $\mathcal{N}$ 的结构 (参见 (18.3.6)), 下面的结论是显见的:

命题 18.4.3

$$\mathcal{N}^{\perp} = \operatorname{span}\left(\Theta_j^i \mid j=2,\cdots,k_i;\ i=1,\cdots,n\right). \tag{18.4.9}$$

18.5 $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的正交分解

文献 [34] 不仅考虑了 $\mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的直交分解 (18.1.3), 也考虑了相应的正交分解. 本节给出正交分解的简单数值方法.

首先注意到, 势博弈可表示为

$$\mathcal{G}_P = \operatorname{span}(B_P) = \operatorname{span}(B_P^0), \tag{18.5.1}$$

并且, $\operatorname{Col}(B_P^0)$ 是 $\mathcal{G}_P$ 的一个基底. 这里, B_P 和 B_P^0 分别由 (18.3.10) 及 (18.3.11) 定义.

非策略子空间可表示为

$$\mathcal{N} = \operatorname{span}(B_N), \tag{18.5.2}$$

并且 $\operatorname{Col}(B_N)$ 是 $\mathcal{N}$ 的基底. B_N 由 (18.3.6) 定义.

利用 (18.5.1) 及 (18.5.2), 定义纯势子空间 $\mathcal{P}$, 纯调和子空间 $\mathcal{H}$, 以及调和博弈子空间 $\mathcal{G}_H$, 如下:

定义 18.5.1 1) 纯势子空间 $\mathcal{P}$ 定义为

$$\mathcal{P} := \mathcal{N}^{\perp} \bigcap \mathcal{G}_{\mathcal{P}}; \tag{18.5.3}$$

2) 纯调和子空间 $\mathcal{H}$ 定义为

$$\mathcal{H} := \mathcal{G}_{\mathcal{P}}^{\perp}. \tag{18.5.4}$$

3) 调和博弈子空间 $\mathcal{G}_H$ 定义为

$$\mathcal{G}_H := \mathcal{N} \uplus \mathcal{H}. \tag{18.5.5}$$

注 在直和分解中, 纯势与纯调和子空间的定义不唯一. 在此后的讨论中, 假定 $\mathcal{P}$ 和 $\mathcal{H}$ 由定义 18.5.1 确定, 则它们有唯一性. 并且, 利用这两个子空间, 分解式 (18.1.3) 变为

$$\mathcal{G}_{[n;k_1,\cdots,k_n]} = \underbrace{\mathcal{P} \oplus \overbrace{\mathcal{N}}_{}}_{\text{势博弈}} \oplus \mathcal{H}, \quad \overbrace{\mathcal{N} \oplus \mathcal{H}}^{\text{调和博弈}} \tag{18.5.6}$$

这里

$$\mathcal{G}_P = \mathcal{P} \oplus \mathcal{N}; \tag{18.5.7}$$

$$\mathcal{G}_H = \mathcal{H} \oplus \mathcal{N}. \tag{18.5.8}$$

(18.5.6) 称为 G 的正交分解.

利用 $\mathcal{G}_P$ 和 $\mathcal{N}$ 的基底, 一个有限博弈 $G \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的正交分解很容易计算. 我们给出如下公式

定理 18.5.1 设 $G \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$ 的结构向量为

$$V^c = (V_1^c, V_2^c, \cdots, V_n^c).$$

那么

1) 它在 $\mathcal{G}_P$ 上的 (正交) 投影为

$$V_{\mathcal{G}_P}^{\mathrm{T}} = B_P^0 \left((B_P^0)^{\mathrm{T}} B_P^0\right)^{-1} (B_P^0)^{\mathrm{T}} (V^c)^{\mathrm{T}}; \tag{18.5.9}$$

2) 它在 $\mathcal{N}$ 上的 (正交) 投影为

$$V_{\mathcal{N}}^{\mathrm{T}} = B_N \left(B_N^{\mathrm{T}} B_N\right)^{-1} B_N^{\mathrm{T}} (V^c)^{\mathrm{T}}; \tag{18.5.10}$$

3) 它在 $\mathcal{H}$ 上的 (正交) 投影为

$$V_{\mathcal{H}} = V^c - V_{\mathcal{G}_P}; \tag{18.5.11}$$

4) 它在 $\mathcal{P}$ 上的 (正交) 投影为

$$V_{\mathcal{P}} = V_{\mathcal{G}_P} - V_{\mathcal{N}}; \tag{18.5.12}$$

5) 它在 $\mathcal{G}_H$ 上的 (正交) 投影为

$$V_{\mathcal{G}_H} = V_{\mathcal{N}} + V_{\mathcal{H}}. \tag{18.5.13}$$

注 1) 如果在 $\mathbb{R}^{nk}$ 上的内积定义为

$$\langle X, Y \rangle := X^{\mathrm{T}} Q Y, \tag{18.5.14}$$

这里 $Q \in \mathcal{M}_{kn \times kn}$ 为一正定矩阵, 那么, 公式 (18.5.9) 变为

$$V_{\mathcal{G}_P}^{\mathrm{T}} = B_P^0 \left((B_P^0)^{\mathrm{T}} Q B_P^0\right)^{-1} (B_P^0)^{\mathrm{T}} Q (V^c)^{\mathrm{T}}; \tag{18.5.15}$$

公式 (18.5.10) 变为

$$V_{\mathcal{N}}^{\mathrm{T}} = B_N \left(B_N^{\mathrm{T}} Q B_N\right)^{-1} B_N^{\mathrm{T}} Q (V^c)^{\mathrm{T}}. \tag{18.5.16}$$

公式 (18.5.11)–(18.5.13) 保持不变.

2) 在文献 [34] 中内积由 (18.5.14) 定义, 其中

$$Q = \operatorname{diag}\left(\underbrace{k_1, \cdots, k_1}_{k}, \underbrace{k_2, \cdots, k_2}_{k}, \cdots, \underbrace{k_n, \cdots, k_n}_{k}\right).$$

下面讨论几个例子.

例 18.5.1　1) 考察一个博弈 $G \in \mathcal{G}_{[2;2,2]}$. 其支付矩阵见表 18.5.1.

表 18.5.1　例 18.5.1 中 1 的支付双矩阵

$P_1\backslash P_2$	1	2
1	a_1, b_1	a_2, b_2
2	a_3, b_3	a_4, b_4

那么, 有

$$V^c = [a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4].$$

$$E_1 = \mathbf{1}_2 \otimes I_2; \qquad E_2 = I_2 \otimes \mathbf{1}_2.$$

$$B_P^0 = \begin{bmatrix} I_4 & E_1 & 0 \\ I_4 & 0 & E_2^0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

$$B_N = \begin{bmatrix} E_1 & 0 \\ 0 & E_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

(i) 考虑性别大战, 则有 $a_1=2$, $a_2=0$, $a_3=0$, $a_4=1$, $b_1=1$, $b_2=0$, $b_3=0$, $b_4=2$[79]. 利用公式 (18.5.9)–(18.5.13) 可得

$$\begin{aligned}
V_{\mathcal{H}} &= [0,0,0,0,0,0,0,0];\\
V_{\mathcal{N}} &= [1,0.5,1,0.5,0.5,0.5,1,1];\\
V_{\mathcal{P}} &= [1,-0.5,-1,0.5,0.5,-0.5,-1,1];\\
V_{\mathcal{G}_P} &= [2,0,0,1,1,0,0,2];\\
V_{\mathcal{G}_H} &= [1,0.5,1,0.5,0.5,0.5,1,1].
\end{aligned}$$

显然它是个势博弈.

(ii) 考虑囚徒困境, 则有 $a_1=-1$, $a_2=-5$, $a_3=0$, $a_4=-4$, $b_1=-1$, $b_2=0$, $b_3=-5$, $b_4=-4$[79]. 类似地可计算

$$\begin{aligned}
V_{\mathcal{H}} &= [0,0,0,0,0,0,0,0];\\
V_{\mathcal{N}} &= [-0.5,-4.5,-0.5,-4.5,-0.5,-0.5,-4.5,-4.5];\\
V_{\mathcal{P}} &= [-0.5,-0.5,0.5,0.5,-0.5,0.5,-0.5,0.5];\\
V_{\mathcal{G}_P} &= [-1,-5,0,-4,-1,0,-5,-4];\\
V_{\mathcal{G}_H} &= [-0.5,-4.5,-0.5,-4.5,-0.5,-0.5,-4.5,-4.5].
\end{aligned}$$

它显然也是个势博弈.

2) 考虑石头 (R)-剪刀 (S)-布 (P) 游戏 $G\in\mathcal{G}_{[2;3,3]}$, 其支付矩阵见表 18.5.2.

表 18.5.2 例 18.5.1 (2) (石头–剪刀–布的支付双矩阵)

$P_1\backslash P_2$	R	S	P
R	0, 0	1, −1	−1, 1
S	−1, 1	0, 0	1, −1
P	1, −1	−1, 1	0, 0

由支付表不难得到

$$\begin{aligned}
V^c = [&0,1,-1,-1,0,1,1,-1,0,\\
&0,-1,1,1,0,-1,-1,1,0].
\end{aligned}$$

$$E_1=\mathbf{1}_3\otimes I_3=\begin{bmatrix} I_3\\ I_3\\ I_3\end{bmatrix};$$

$$E_2=I_3\otimes\mathbf{1}_3=\begin{bmatrix} \mathbf{1}_3 & 0 & 0\\ 0 & \mathbf{1}_3 & 0\\ 0 & 0 & \mathbf{1}_3\end{bmatrix}.$$

于是, 可算出

$$B_N = \begin{bmatrix} I_3 & 0 & 0 & 0 \\ I_3 & 0 & 0 & 0 \\ I_3 & 0 & 0 & 0 \\ 0 & \mathbf{1}_3 & 0 & 0 \\ 0 & 0 & \mathbf{1}_3 & 0 \\ 0 & 0 & 0 & \mathbf{1}_3 \end{bmatrix}; \quad B_N^0 = \begin{bmatrix} I_3 & 0 & 0 \\ I_3 & 0 & 0 \\ I_3 & 0 & 0 \\ 0 & \mathbf{1}_3 & 0 \\ 0 & 0 & \mathbf{1}_3 \\ 0 & 0 & 0 \end{bmatrix}.$$

$$B_P^0 = \left[\begin{pmatrix} I_9 \\ I_9 \end{pmatrix} \ B_{\mathcal{N}}^0 \right].$$

利用公式 (18.5.9)–(18.5.13), 有

$$\begin{aligned} V_{\mathcal{G}_P} = V_{\mathcal{N}} &= V_{\mathcal{P}} = 0; \\ V_{\mathcal{H}} = V_{\mathcal{G}_H} &= [0,1,-1,-1,0,1,1,-1,0, \\ &\qquad 0,-1,1,1,0,-1,-1,1,0]. \end{aligned}$$

这是一个纯调和博弈.

18.6 应用举例

18.6.1 近似势博弈的收敛性

由于势博弈有许多优点, 特别是演化势博弈在 MBRA 更新规则下会收敛到纯纳什均衡点, 一个自然的问题就是, 如果一个博弈很接近势博弈, 那它所对应的演化博弈会有这种性质吗? 这是一个很自然而有趣的问题. 先回忆一下演化博弈, 它的动态过程可以用局势演化方程来刻画. 举一例子:

例 18.6.1 ([35])　一个有限博弈 $G \in \mathcal{G}_{[2;3,3]}$, 这里 $S_1 = S_2 = \{1,2,3\}$: 1: 工作; 2: 在办公室偷懒; 3: 在家偷懒. 其支付矩阵见表 18.6.1.

表 18.6.1　例 18.6.1 的支付矩阵

支付 \ 局势	11	12	13	21	22	23	31	32	33
c_1	90	−12	48	−12	1	24	48	24	1
c_2	90	−12	48	−12	−1	24	48	24	−1

设策略更新规则为 MBRA, 则不难找到各种情况下的最优反应策略, 见表 18.6.2. (因为 $|S_1^{-1}| = |S_2^{-1}| = 1$, D-MBRA 和 S-MBRA 没有区别.)

表 18.6.2 例 18.6.1 最优反应策略

$(t+1)$ 时局势 \ t 时局势	11	12	13	21	22	23	31	32	33
f_1	1	3	1	1	3	1	1	3	1
f_2	1	1	1	3	3	3	1	1	1

于是有

$$x_i(t+1) = f_i(x_1(t), x_2(t)) = M_i x(t), \quad i = 1, 2, \tag{18.6.1}$$

这里 $x(t) = \ltimes_{i=1}^2 x_i(t)$, M_i 是 f_i 的结构矩阵, $i = 1, 2$, 它们是

$$\begin{aligned} M_1 &= \delta_3[1,3,1,1,3,1,1,3,1]; \\ M_2 &= \delta_3[1,1,1,3,3,3,1,1,1]. \end{aligned} \tag{18.6.2}$$

定义 18.6.1 两个演化博弈称为动态等价的, 如果它们的局势演化方程相同.

下面的两个动态等价的性质可由直接检验证明.

命题 18.6.1 如果两个博弈策略等价, 那么 (在相同的策略演化规则下) 它们也是动态等价的.

命题 18.6.2 如果一个博弈 G 与一个势博弈 G_P 动态等价, 那么, 在 MBRA 下 G 也将收敛到纯纳什均衡点.

证明 由于 G_P 收敛到一个纳什均衡点 $x^* \in S$, G 也将收敛到 x^*. 这时如果 x^* 不是 G 的一个纯纳什均衡点, 那么, 至少存在一个 i 使得

$$c_i\left(x_i, (x^*)^{-i}\right) > c_i\left(x_i^*, (x^*)^{-i}\right).$$

于是, 当 i 被选更新时, 玩家 i 将从 x_i^* 运动到 x_i, 这违背了 G 与 G_P 动态等价的假定. □

注 1) 有许多策略更新规则会导致一个演化势博弈收敛到一个纯纳什均衡点. 那么显然, 如果 G 和离它最近的势博弈 G_P 动态等价, 则 G 也会收敛到一个纯纳什均衡点.

2) 一个级联式的演化势博弈

$$\begin{cases} x_1(t+1) = f_1(x_1(t), x_2(t), \cdots, x_n(t)), \\ x_2(t+1) = f_2(x_1(t+1), x_2(t), \cdots, x_n(t)), \\ \qquad \vdots \\ x_n(t+1) = f_1(x_1(t+1), \cdots, x_{n-1}(t+1), x_n(t)), \end{cases}$$

即使所有玩家同时用 MBRA 更新自己策略, 它也会收敛到一个纯纳什均衡点.

3) 对于演化势博弈, 许多学习算法都会收敛到一个纯纳什均衡点. 例如, 虚拟行为 (fictitious play)[112], 对数线性学习算法 (log-linear learning)[27] 及其改进形式[110] 等. 当 G 与 G_P 动态等价时, 它们均能保证 G 收敛到一个纯纳什均衡点.

例 18.6.2 回顾例 18.6.1. 利用公式 (18.5.9) 可以算出该例中 G 在 $\mathcal{G}_P$ 上的投影:

$$
\begin{aligned}
V_{\mathcal{G}_P} = [\,&89.7778, -11.8889, 48.1111, -11.8889,\\
&0.4444, 24.4444, 48.1111, 24.4444, 0.4444,\\
&90.2222, -12.1111, 47.8889, -12.1111, -0.4444,\\
&23.5556, 47.8889, 23.5556, -0.4444].
\end{aligned}
$$

于是, 容易算出 G_P 的局势演化方程与 (18.6.1)–(18.6.2) 相同. 依据命题 18.6.2 可知, 利用不同步 MBRA (或者前面注释中提到的任何方法), G 都将收敛到一个纯纳什均衡点.

18.6.2 网络演化博弈的分解

先回忆一下第 16 章中讨论过的网络演化博弈.

定义 18.6.2 一个网络演化博弈, 记作 $\mathcal{G} = ((N, E), G, \Pi)$, 由三个要素组成:

(i) 网络图: (N, E);

(ii) 基本网络博弈: G 是两个玩家的游戏. 玩家 i 和 j 玩该游戏, 当且仅当, 它们有边相联, 即 $(i, j) \in E$.

(iii) 基于局部信息的策略更新规则:

$$
x_i(t+1) = f_i(x_j(t), c_j(t) \mid j \in U(i)), \quad i = 1, \cdots, n. \tag{18.6.3}
$$

记

$$
U_s(i) := \left\{k \mid \text{存在一条路径联接 } i \text{ 和 } k, \text{且其长度} \leqslant s\right\}.
$$

那么, (18.6.3) 可表示为

$$
x_i(t+1) = f_i(x_j(t) \mid j \in U_2(i)), \quad i = 1, \cdots, n. \tag{18.6.4}
$$

第 17 章证明了以下结论:

定理 18.6.1 给定一个网络演化博弈 $\mathcal{G} = ((N, E), G, \Pi)$, 如果基本网络博弈 G 是势博弈, 则 $\mathcal{G}$ 也是势博弈. 并且, 设 $e = (i, j) \in E$ 为一条边, 且 i 与 j 之间的博弈势为 P_e. 则网络演化博弈 $\mathcal{G}$ 的势为

$$
P_{\mathcal{G}} = \sum_{e \in E} P_e. \tag{18.6.5}
$$

下面给出非策略博弈的一个性质:

命题 18.6.3 $G \in \mathcal{N}$, 当且仅当, G 是势博弈且其势函数为零 (或为任意常数).

证明 (必要性) 因为 $\mathcal{N} \subset \mathcal{G}_P$, G 显然是势博弈. 并且, 由定义

$$(V^c)^{\mathrm{T}} \in \operatorname{span}(B_N). \tag{18.6.6}$$

根据 (18.3.6), (18.6.6) 等价于存在 $\xi_i \in \mathbb{R}^{k/k_i}$, $i = 1, \cdots, n$ 使得

$$V_i^c = \xi_i^{\mathrm{T}} E_i^{\mathrm{T}}. \tag{18.6.7}$$

于是有

$$\begin{aligned} c_i(x_1, \cdots, x_n) &= V_i^c \ltimes_{j=1}^n x_j \\ &= \xi_i^{\mathrm{T}} \left(I_{k[1,i-1]} \otimes \mathbf{1}_{k_i}^{\mathrm{T}} \otimes I_{k[i+1,n]} \right) \ltimes_{j=1}^n x_j \\ &= \xi_i^{\mathrm{T}} \ltimes_{j \neq i} x_j, \quad i = 1, \cdots, n. \end{aligned}$$

因此, c_i 与 x_i 无关. 于是, 势函数 $P(x_i, \cdots, x_n)$ 也与 x_i 无关. 但 i 是任意的, 因此, $P(x_i, \cdots, x_n) = \text{const}$.

(充分性) 设 G 有零势函数. 则 c_i 与 x_i 无关, $\forall i$. 因此存在 $\xi_i \in \mathbb{R}^{k/k_i}$ 使得

$$\begin{aligned} c_i(x_1, \cdots, x_n) &= \xi_i^{\mathrm{T}} \ltimes_{j \neq i} x_j \\ &= \xi_i^{\mathrm{T}} \left(I_{k[1,i-1]} \otimes \mathbf{1}_{k_i}^{\mathrm{T}} \otimes I_{k[i+1,n]} \right) \ltimes_{j=1}^n x_j \\ &= \xi_i^{\mathrm{T}} E_i^{\mathrm{T}} \ltimes_{j=1}^n x_j \\ &= V_i^c \ltimes_{j=1}^n x_j, \quad i = 1, \cdots, n. \end{aligned}$$

即

$$(V_i^c)^{\mathrm{T}} = E_i \xi_i, \quad i = 1, \cdots, n.$$

因此

$$B_N \xi = (V^c)^{\mathrm{T}},$$

这表明 $G \in \mathcal{N}$. □

命题 18.6.4 考察一个网络演化博弈 $\mathcal{G} = ((N, E), G, \Pi)$. 如果基本网络博弈 $G \in \mathcal{N}$, 则 $\mathcal{G} \in \mathcal{N}$, 即, 网络演化博弈 $\mathcal{G}$ 也是非策略的.

证明 因为 G 有定常势函数, 利用 (18.6.5), 网络演化博弈 $\mathcal{G}$ 的势函数也是定常的. 根据命题 18.6.3, $\mathcal{G} \in \mathcal{N}$. □

由于 $\mathcal{N}$, $\mathcal{G}_P$ 和 $\mathcal{H}$ 之间的线性关系, 我们有如下推论:

推论 18.6.1 考察一个网络演化博弈 $\mathcal{G} = ((N, E), G, \Pi)$. 如果 $G \in \mathcal{G}_P$(或者 $G \in \mathcal{H}$, $G \in \mathcal{P}$, $G \in \mathcal{G}_H$), 那么 $\mathcal{G} \in \mathcal{G}_P$ (或者, 对应地, $\mathcal{G} \in \mathcal{H}$, $\mathcal{G} \in \mathcal{P}$, $\mathcal{G} \in \mathcal{G}_H$).

利用推论 18.6.1, 我们可以得到以下网络演化博弈的分解定理.

推论 18.6.2 考察一个网络演化博弈 $\mathcal{G}=((N,E),G,\Pi)$. 它具有如下的正交分解:

$$
\mathcal{G}=\underbrace{\mathcal{P}\oplus\overbrace{\mathcal{N}}^{}}_{\text{势博弈}}\overbrace{\oplus\ \mathcal{H}}^{\text{调和博弈}}, \tag{18.6.8}
$$

这里

$$
\begin{aligned}
&\mathcal{P}=\sum_{e\in E}\mathcal{P}^e;\quad \mathcal{N}=\sum_{e\in E}\mathcal{N}^e;\\
&\mathcal{H}=\sum_{e\in E}\mathcal{H}^e;\\
&\mathcal{G}_P=\sum_{e\in E}\mathcal{G}_P^e;\quad \mathcal{G}_H=\sum_{e\in E}\mathcal{G}_H^e,
\end{aligned}
$$

并且, $\mathcal{P}^e$, $\mathcal{N}^e$, $\mathcal{H}^e$, $\mathcal{G}_P^e$, 以及 $\mathcal{G}_H^e$ 为基本网络博弈 G 在边 $e\in E$ 上的相应分解.

我们用一个例子来描述这种分解.

例 18.6.3 考察一个网络演化博弈 $\mathcal{G}=((N,E),G,\Pi)$, 这里 (i) 网络图为图 18.6.1; (ii) 基本网络博弈 G 为 Benoit-Krishna 游戏[123]. 注意, Benoit-Krishna 游戏有

$$
S_0=\{1(D):\ \text{抵赖},\ \ 2(W):\ \text{胡说八道},\ \ 3(C):\ \text{招供}\}.
$$

支付矩阵见表 18.6.3; (iii) 本例不考虑策略更新规则.

表 18.6.3　支付双矩阵 (Benoit-Krishna 游戏)

$P_1\backslash P_2$	$D=1$	$W=2$	$C=3$
$D=1$	(10, 10)	(−1, −12)	(−1, 15)
$W=2$	(−12, −1)	(8, 8)	(−1, −1)
$C=3$	(15, −1)	(−1, −1)	(0, 0)

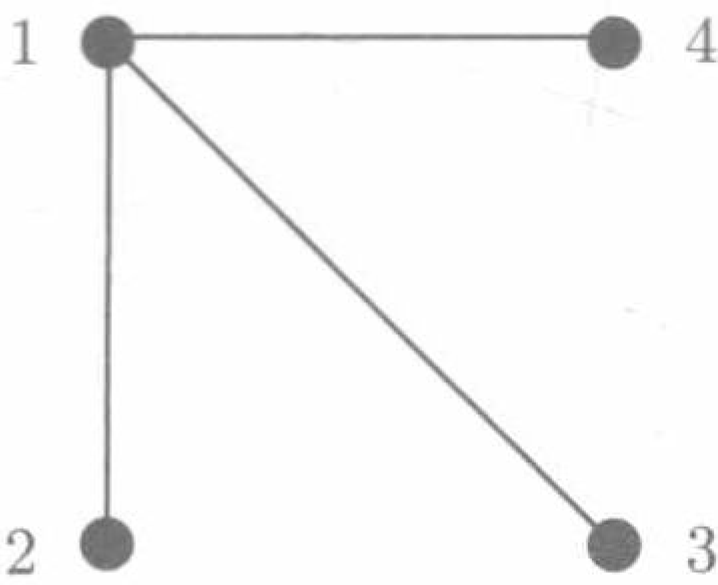


图 18.6.1　例 18.6.3 的网络图

我们用两种方法计算网络演化博弈 $\mathcal{G}$ 的分解.

1) 逐边计算 (计算基本网络博弈 G 在每条边 $e\in E$ 上的分解, 再将各分量相加).

注意到 Benoit-Krishna 游戏 $G \in \mathcal{G}_{[2;3,3]}$, 与例 18.5.1 中石头–剪刀–布属同一空间, 因此, 例 18.5.1 中 B_P (B_P^0) 以及 B_N 可直接用来计算投影.

G 的结构向量为

$$V^c = [10, -1, -1, -12, 8, -1, 15, -1, 0,\\ 10, -12, 15, -1, 8, -1, -1, -1, 0].$$

类似石头–剪刀–布, 可以算得

$$V_{\mathcal{H}} = [0, 4.5, -4.5, -4.5, 0, 4.5, 4.5, -4.5, 0,\\ 0, -4.5, 4.5, 4.5, 0, -4.5, -4.5, 4.5, 0];$$

$$V_{\mathcal{N}} = [4.3333, 2, -0.6667, 4.3333, 2, -0.6667, 4.3333, 2, -0.6667,\\ 4.3333, 4.3333, 4.3333, 2, 2, 2, -0.6667, -0.6667, -0.6667];$$

$$V_{\mathcal{P}} = [5.6667, -7.5, 4.1667 - 11.8333, 6, -4.8333, 6.1667, 1.5, 0.6667,\\ 5.6667, -11.8333, 6.1667, -7.5, 6, 1.5, 4.1667, -4.8333, 0.6667];$$

$$V_{\mathcal{G}_P} = [10, -5.5, 3.5, -7.5, 8, -5.5, 10.5, 3.5, 0,\\ 10, -7.5, 10.5, -5.5, 8, 3.5, 3.5, -5.5, 0];$$

$$V_{\mathcal{G}_H} = [4.3333, 6.5, -5.1667, -0.1667, 2, 3.8333, 8.8333, -2.5, -0.6667,\\ 4.3333, -0.1667, 8.8333, 6.5, 2, -2.5, -5.1667, 3.8333, -0.6667].$$

现在将 $V_{\mathcal{H}}$ 分为相等的两个部分如下:

$$V_{\mathcal{H}} = \left[V_{\mathcal{H}}^1, V_{\mathcal{H}}^2\right],$$

这里 $V_{\mathcal{H}}^1$ 和 $V_{\mathcal{H}}^2$ 分别为玩家 1, 2 的支付函数结构向量 V_1^c 和 V_2^c 的投影.

下面计算基本网络博弈 G 在边 $(1,2)$ 上的分解. V_1^c 及 V_2^c 在 $\mathcal{H}$ 上的投影为

$$H_1^1(x_1, x_2) = V_{\mathcal{H}}^1 x_1 x_2 = V_{\mathcal{H}}^1 \left(I_9 \otimes \mathbf{1}_9^{\mathrm{T}}\right) \ltimes_{i=1}^4 x_i;\\ H_2^1(x_1, x_2) = V_{\mathcal{H}}^2 x_1 x_2 = V_{\mathcal{H}}^2 \left(I_9 \otimes \mathbf{1}_9^{\mathrm{T}}\right) \ltimes_{i=1}^4 x_i.$$

类似地, 对于 G 在边 $(1,3)$ 上的分解, 有

$$\begin{aligned} H_1^2(x_1, x_3) &= V_{\mathcal{H}}^1 x_1 x_3\\ &= V_{\mathcal{H}}^1 \left(I_3 \otimes \mathbf{1}_3^{\mathrm{T}} \otimes I_3 \otimes \mathbf{1}_3^{\mathrm{T}}\right) \ltimes_{i=1}^4 x_i;\\ H_3^2(x_1, x_3) &= V_{\mathcal{H}}^2 x_1 x_3\\ &= V_{\mathcal{H}}^2 \left(I_3 \otimes \mathbf{1}_3^{\mathrm{T}} \otimes I_3 \otimes \mathbf{1}_3^{\mathrm{T}}\right) \ltimes_{i=1}^4 x_i. \end{aligned}$$

对于 G 在边 $(1,4)$ 上的分解, 有

$$H_1^3(x_1, x_4) = V_{\mathcal{H}}^1 x_1 x_4 = V_{\mathcal{H}}^1 \left(I_3 \otimes \mathbf{1}_9^{\mathrm{T}} \otimes I_3\right) \ltimes_{i=1}^4 x_i;\\ H_4^3(x_1, x_4) = V_{\mathcal{H}}^2 x_1 x_4 = V_{\mathcal{H}}^2 \left(I_3 \otimes \mathbf{1}_9^{\mathrm{T}} \otimes I_3\right) \ltimes_{i=1}^4 x_i.$$

最后, 网络演化博弈 $\mathcal{G}$ 在纯调和子空间 $\mathcal{H}$ 上的分解为

$$V_{\mathcal{H}} = [H_1, H_2, H_3, H_4],$$

这里

$$H_1 = H_1^1 + H_1^2 + H_1^3;\ H_2 = H_2^1;\ H_3 = H_3^2;\ H_4 = H_4^3.$$

最后, 数值结果为

$$V_{\mathcal{H}} = [0, 4.5, -4.5, 4.5, 9, \cdots, 4.5, 0, -4.5, 4.5, 0] \in \mathbb{R}^{324}. \tag{18.6.9}$$

类似计算可以找到网络演化博弈 $\mathcal{G}$ 在 $\mathcal{N}$, $\mathcal{P}$, $\mathcal{G}_P$, 以及 $\mathcal{G}_H$ 上的投影, 它们分别为

$$\begin{aligned} V_{\mathcal{N}} = &[13, 10.6667, 8, 10.6667, 8.3333, \cdots, \\ &-0.6667, -0.6667, -0.6667, -0.6667, -0.6667] \in \mathbb{R}^{324}. \end{aligned} \tag{18.6.10}$$

$$\begin{aligned} V_{\mathcal{P}} = &[17, 3.8333, 15.5, 3.8333, -9.3333, \cdots, \\ &-4.8333, 0.6667, 4.1667, -4.8333, 0.6667] \in \mathbb{R}^{324}. \end{aligned} \tag{18.6.11}$$

$$V_{\mathcal{G}_P} = [30, 14.5, 23.5, 14.5, -1, \cdots, -5.5, 0, 3.5, -5.5, 0] \in \mathbb{R}^{324}. \tag{18.6.12}$$

$$\begin{aligned} V_{\mathcal{G}_H} = &[13, 15.1667, 3.5, 15.1667, 17.3333, \cdots \\ &3.8333, -0.6667, -5.1667, 3.8333, -0.6667] \in \mathbb{R}^{324}. \end{aligned} \tag{18.6.13}$$

2) 全局计算 (考虑网络演化博弈为一整体多人博弈, 直接计算 $\mathcal{G}$ 的投影). 首先, 不难算出整体支付向量, 它是

$$V_{\mathcal{G}} = [30, 19, 19, 19, 8, \cdots, -1, 0, -1, -1, 0] \in \mathbb{R}^{324}.$$

利用

$$\begin{aligned} &\Psi_1 = \mathbf{1}_3 \otimes I_{27}, \\ &\Psi_2 = I_3 \otimes \mathbf{1}_3 \otimes I_9, \\ &\Psi_3 = I_9 \otimes \mathbf{1}_3 \otimes I_3, \\ &\Psi_4 = I_{27} \otimes \mathbf{1}_3, \end{aligned}$$

不难算出 B_N 和 B_P^0. 最后, 利用公式 (18.5.9)–(18.5.13) 可直接算出各种投影. 数值结果显示, 各投影与 (18.6.9)–(18.6.13) 中的结果一致.

18.7 注释与参考

由于势博弈的重要性, 人们不仅关心它所在有限博弈空间中的位置, 也关心它的邻域. 关于有限博弈的向量空间结构, 最早是由 Candogan, Menache 等提出的[34]. 利用向量空间结构, 他们也讨论了近似势博弈等[35]. 文献 [34] 用算子的方法定义了 3 个基本子空间如下:

$$
\begin{aligned}
&\mathcal{P} := \left\{ u \in \mathbb{R}^{kn} \mid u = \Pi u, \text{ 且 } Du \in \mathrm{im}\delta_0 \right\}; \\
&\mathcal{H} := \left\{ u \in \mathbb{R}^{kn} \mid u = \Pi u, \text{ 且 } Du \in \ker \delta_0^* \right\}; \\
&\mathcal{N} := \left\{ u \in \mathbb{R}^{kn} \mid u \in \ker D \right\},
\end{aligned}
\tag{18.7.1}
$$

这里 δ_0, Π, D 为一组组合算子. 然后, 利用代数拓扑中的一个经典结论: Helmholtz 分解定理[137], 证明了分解式 (18.1.3).

本章利用代数状态空间方法不仅大大简化了他们的方法, 而且, 使子空间具有更清晰的物理意义.

第19章 对称博弈

19.1 玩家对称博弈

考察一个有限博弈 $G \in \mathcal{G}_{[n;k_1,\cdots,k_n]}$. 为了研究它的对称性, 本章总假定

A1:

$$|S_i| = \kappa, \quad i = 1, \cdots, n.$$

于是, 简记 $\mathcal{G}_{[n;\kappa]} := \mathcal{G}_{[n;\kappa,\cdots,\kappa]}$.

定义 19.1.1 给定 $G \in \mathcal{G}_{[n,\kappa]}$ 以及 $\sigma \in \mathbf{S}_n$. G 称为关于 σ 对称的, 如果

$$c_i(x_1, \cdots, x_n) = c_{\sigma(i)}\left(x_{\sigma(1)}, \cdots, x_{\sigma(n)}\right), \quad i = 1, \cdots, n. \tag{19.1.1}$$

根据定义, 下面的结论是显见的:

命题 19.1.1 设 $G \in \mathcal{G}_{[n,\kappa]}$. 定义 G 的对称群

$$\Pi(G) := \{\sigma \in \mathbf{S}_n | G\text{关于 } \sigma \text{ 对称}\}. \tag{19.1.2}$$

那么 $\Pi(G) < \mathbf{S}_n$, 即 $\Pi(G)$ 是 $\mathbf{S}_n$ 的子群.

定义 19.1.2 ([31]) 设 $G \in \mathcal{G}_{[n,\kappa]}$. G 称为一个玩家对称博弈, 如果

$$\Pi(G) = \mathbf{S}_n.$$

定义 19.1.3 设 $G \in \mathcal{G}_{[n,\kappa]}$. 一局势 $x = (x_1, \cdots, x_n) \in S$ 给定. x 的策略指标 (action multiplicity), 记作 $\#(x)$, 如下定义

$$\#(x) = (\#(1, x), \#(2, x), \cdots, \#(\kappa, x)) \in \mathbb{Z}_+^\kappa, \tag{19.1.3}$$

这里 $\#(j, x)$ 是局势 x 中取策略 j 的玩家的数目, 即

$$\#(j, x) = |\{x_r | x_r = j, \ 1 \leqslant r \leqslant n\}|.$$

为讨论方便, 先给出对称群 $\mathbf{S}_n$ 的一些简单的结构性质. 其证明可通过一些初等计算直接验证, 亦可见文献 [2, 5].

引理 19.1.1 $\mathbf{S}_n$ 可由对换生成, 即

$$\mathbf{S}_n = \langle (i, j) | 1 \leqslant i < j \leqslant n \rangle. \tag{19.1.4}$$

下面的性质是基本的.

命题 19.1.2 $G \in \mathcal{G}_{[n;\kappa]}$ 是一个玩家对称博弈, 当且仅当

$$c_i(x_i, x^{-i}) = c_j(y_j, y^{-j}), \quad 1 \leqslant i, j \leqslant n, \tag{19.1.5}$$

这里 $x_i = y_j$ 且 $\#(x^{-i}) = \#(y^{-j})$.

证明 (必要性) 构造一个排列 $\sigma \in \mathbf{S}_n$ 满足 $\sigma(i) = j$, 并且, 对于 $x_p \in x^{-i}$, $y_q \in y^{-j}$, 如果 $x_p = y_q$, 则 $\sigma(p) = q$. 由于 $\#(x^{-i}) = \#(y^{-j})$, 则可构造一对一的 σ. 由于博弈是玩家对称的, 则由 (19.1.1) 可得 (19.1.5).

(充分性) 注意到对任何 $\sigma \in \mathbf{S}$, 记 $x^\sigma := \{x_{\sigma(1)}, \cdots, x_{\sigma(n)}\}$, 则 $\#(x) = \#(x^\sigma)$. 令 $\sigma = (i, j)$. 因为 $x_i = y_j$ 则 $\#(x^{-i}) = \#(y^{-j})$. 则由 (19.1.5) 可知 (19.1.1) 对该 σ 成立. 再由引理 19.1.1, (19.1.1) 对任何 $\sigma \in \mathbf{S}_n$ 成立. □

设 $\sigma \in \mathbf{S}_n$ 给定. 我们寻找一个矩阵 $M_\sigma \in \mathcal{M}_{\kappa^n \times \kappa^n}$ 使得

$$M_\sigma \ltimes_{i=1}^n x_i = \ltimes_{i=1}^n x_{\sigma(i)}, \quad x_i \in \Delta_\kappa,\ i = 1, \cdots, n. \tag{19.1.6}$$

这样的 M_σ 称为 σ 的矩阵表示.

引理 19.1.2 ([88]) $\mathbf{S}_n$ 可由对换 $(1, j)$, $1 < j \leqslant n$ 生成, 即

$$\mathbf{S}_n = \langle (1, r) | r = 2, \cdots, n \rangle. \tag{19.1.7}$$

引理 19.1.3 设 $\sigma \in \mathbf{S}_n$. 则存在唯一的 $M_\sigma \in \mathcal{M}_{\kappa^n \times \kappa^n}$, 使得 (19.1.6) 成立.

证明 由引理 19.1.2 可知, 只要对 $\sigma = (1, r)$ 构造出 M_σ 就行了. 不难看出

$$W_{[\kappa^{r-2}, \kappa]} W_{[\kappa, \kappa^{r-1}]} x_1 \cdots x_r \cdots x_n = x_r x_2 \cdots x_{r-1} x_1 \cdots x_n.$$

所以

$$M_\sigma = \left[W_{[\kappa^{r-2}, \kappa]} W_{[\kappa, \kappa^{r-1}]} \right] \otimes I_{\kappa^{n-r}}. \tag{19.1.8}$$

唯一性是显然的. □

利用引理 19.1.3, 可得下述命题:

命题 19.1.3 给定 $G \in \mathcal{G}_{[n;\kappa]}$. $\sigma \in \Pi(G)$, 当且仅当

$$V_i^c = V_{\sigma(i)}^c M_\sigma. \tag{19.1.9}$$

证明 将 (19.1.1) 表示为向量形式如下:

$$V_i^c \ltimes_{i=1}^n x_i = V_{\sigma(i)}^c \ltimes_{i=1}^n x_{\sigma(i)} = V_{\sigma(i)}^c M_\sigma \ltimes_{i=1}^n x_i.$$

(19.1.9) 则显而易见. □

下面讨论一个例子.

例 19.1.1 设 $G \in \mathcal{G}_{[3;3]}$. $\sigma = (1,3)$. G 关于 σ 对称, 当且仅当

$$\begin{cases} V_3^c M_\sigma = V_1^c, \\ V_2^c M_\sigma = V_2^c. \end{cases} \tag{19.1.10}$$

因为

$$\begin{aligned} M_\sigma = W_{[3,3]} W_{[3,9]} = \delta_{27}[&1,10,19,4,13,22,7,16,25,2,11,20,5,14, \\ &23,8,17,26,3,12,21,6,15,24,9,18,27], \end{aligned}$$

不妨记

$$V_1^c = [a_1, a_2, \cdots, a_{27}].$$

那么, 有

$$\begin{aligned} V_3^c = & V_1^c M_\sigma^{-1} \\ = & [a_1, a_{10}, a_4, a_{13}, a_{22}, a_7, a_{16}, a_{25}, \\ & a_2, a_{11}, a_{20}, a_5, a_{14}, a_{23}, a_8, a_{17}, a_{26}, \\ & a_3, a_{12}, a_{21}, a_6, a_{15}, a_{24}, a_9, a_{18}, a_{27}]. \end{aligned}$$

设

$$V_2^c = [b_1, b_2, \cdots, b_{27}].$$

为满足 (19.1.10) 第二个方程, 有

$$\begin{aligned} &b_{10} = b_2, \quad b_{13} = b_5, \quad b_{16} = b_8, \\ &b_{19} = b_3, \quad b_{20} = b_{12}, \quad b_{22} = b_6, \\ &b_{23} = b_{15}, \quad b_{25} = b_9, \quad b_{26} = b_{18}. \end{aligned}$$

设 $G \in \mathcal{G}_{[n;\kappa]}$ 为一玩家对称博弈. 那么有

$$\begin{aligned} c_i(x) = & V_i^c \ltimes_{j=1}^n x_j \\ = & V_i^c W_{[\kappa,\kappa^{i-1}]} x_i \ltimes_{j \neq i} x_j. \end{aligned} \tag{19.1.11}$$

应用 $\sigma = (1,s) \in \mathbf{S}_{n-1}$ ($2 \leqslant s \leqslant n-1$) 于 $\{x_j | j \neq i\}$, 则 $c_i(x)$ 的值不变. 根据引理 19.1.2, 有

$$\begin{aligned} c_i(x) = & V_i^c W_{[\kappa,\kappa^{i-1}]} x_i W_{[\kappa^{s-2},\kappa]} W_{[\kappa,\kappa^{s-1}]} \ltimes_{j=1,\, j \neq i}^n x_j \\ = & V_i^c W_{[\kappa,\kappa^{i-1}]} \left(I_\kappa \otimes W_{[\kappa^{s-2},\kappa]} W_{[\kappa,\kappa^{s-1}]}\right) x_i \ltimes_{j=1,\, j \neq i}^n x_j \\ = & V_i^c W_{[\kappa,\kappa^{i-1}]} \left(I_\kappa \otimes W_{[\kappa^{s-2},\kappa]} W_{[\kappa,\kappa^{s-1}]}\right) W_{[\kappa^{i-1},\kappa]} \ltimes_{j=1}^n x_j. \end{aligned} \tag{19.1.12}$$

比较 (19.1.11) 与 (19.1.12) 可得

$$V_i^c \left[W_{[\kappa,\kappa^{i-1}]} \left(I_\kappa \otimes W_{[\kappa^{s-2},\kappa]} W_{[\kappa,\kappa^{s-1}]}\right) W_{[\kappa^{i-1},\kappa]} - I_r\right] = 0, \quad s = 2, 3, \cdots, n-1, \tag{19.1.13}$$

这里 $r = \max\{\kappa^i, \kappa^{s+1}\}$.

根据引理 19.1.2, 显然 (19.1.13) 是单个 c_i 所应满足的充要条件. 下面讨论 c_i 与 c_j 所要满足的交叉条件. 因为

$$\begin{aligned} c_p(x) &= V_p^c W_{[\kappa,\kappa^{p-1}]} x_p \ltimes_{j \neq p} x_j, \\ c_q(x) &= V_q^c W_{[\kappa,\kappa^{q-1}]} x_q \ltimes_{j \neq q} x_j, \end{aligned} \tag{19.1.14}$$

而且, 只要 $x_p = x_q$ 以及 $\#(\ltimes_{j\neq p} x_j) = \#(\ltimes_{j\neq q} x_j)$ (因子在乘积 $\ltimes_{j\neq p} x_j$ 或 $\ltimes_{j\neq q} x_j$ 中的顺序无影响), 有 $c_p(x) = c_q(x)$. 因此, 可得出

$$V_p^c W_{[\kappa,\kappa^{p-1}]} = V_q^c W_{[\kappa,\kappa^{q-1}]}. \tag{19.1.15}$$

结合 (19.1.13) 及 (19.1.15) 并且, 取 c_1 为参考玩家的支付函数, 则可得到如下结论:

定理 19.1.1 $G \in \mathcal{G}_{[n;\kappa]}$ 是一个玩家对称博弈, 当且仅当

1)

$$V_1^c \left[\Phi_s - I_{k^{s+1}}\right] = 0, \quad s = 2, 3, \cdots, n-1, \tag{19.1.16}$$

这里

$$\Phi_s := I_\kappa \otimes \left(W_{[\kappa^{s-2},\kappa]} W_{[\kappa,\kappa^{s-1}]}\right); \tag{19.1.17}$$

2)

$$V_i^c = V_1^c W_{[\kappa^{i-1},\kappa]}, \quad i = 2, 3, \cdots, n. \tag{19.1.18}$$

例 19.1.2 考察一个博弈 $G \in \mathcal{G}_{[n;\kappa]}$.

1) 设 $n = 2$: 注意, 当 $n = 2$ 时, 式 (19.1.16) 不存在. 因此, $V_1^c \in \mathbb{R}^{k^2}$ 可以任选. 记

$$V_1^c = [a_{1,1}\ a_{1,2}\ \cdots\ a_{1,\kappa};\ \cdots;\ a_{\kappa,1}\ a_{\kappa,2}\ \cdots\ a_{\kappa,\kappa}].$$

那么有

$$V_2^c = V_1^c W_{[\kappa,\kappa]} = [a_{1,1}\ a_{2,1}\ \cdots\ a_{\kappa,1};\ \cdots;\ a_{1,\kappa}\ a_{2,\kappa}\ \cdots\ a_{\kappa,\kappa}].$$

这个结果很自然, 也是熟知的.

2) 设 $n = 3$: 那么 (19.1.16) 变为

$$V_1^c \left(I_\kappa \otimes W_{[\kappa,\kappa]} - I_{\kappa^3}\right) = 0.$$

- 设 $\kappa=2$, 则有

$$\Phi_2 - I_8 = I_2 \otimes W_{[2,2]} - I_8 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

容易算得

$$V_1^c = [a_1\ a_2\ a_2\ a_3\ a_4\ a_5\ a_5\ a_6].$$

利用 (19.1.18), 有

$$\begin{aligned} V_2^c &= V_1^c W_{[2,2]} = [a_1\ a_2\ a_4\ a_5\ a_2\ a_3\ a_5\ a_6], \\ V_3^c &= V_1^c W_{[4,2]} = [a_1\ a_4\ a_2\ a_5\ a_2\ a_5\ a_3\ a_6]. \end{aligned}$$

- 设 $\kappa=3$, 利用 (19.1.16), 有

$$\begin{aligned} V_1^c = [a_1 \quad & a_2 \quad a_3 \quad a_2 \quad a_4 \quad a_5 \quad a_3 \quad a_5 \quad a_6 \\ & a_7 \quad a_8 \quad a_9 \quad a_8 \quad a_{10} \quad a_{11} \quad a_9 \quad a_{11} \quad a_{12} \\ & a_{13} \quad a_{14} \quad a_{15} \quad a_{14} \quad a_{16} \quad a_{17} \quad a_{15} \quad a_{17} \quad a_{18}]. \end{aligned}$$

用这个 V_1^c 以及公式 (19.1.18) 可得

$$\begin{aligned} V_2^c = [a_1 \quad & a_2 \quad a_3 \quad a_7 \quad a_8 \quad a_9 \quad a_{13} \quad a_{14} \quad a_{15} \\ & a_2 \quad a_4 \quad a_5 \quad a_8 \quad a_{10} \quad a_{11} \quad a_{14} \quad a_{16} \quad a_{17} \\ & a_3 \quad a_5 \quad a_6 \quad a_9 \quad a_{11} \quad a_{12} \quad a_{15} \quad a_{17} \quad a_{18}]; \end{aligned}$$

$$\begin{aligned} V_3^c = [a_1 \quad & a_7 \quad a_{13} \quad a_2 \quad a_8 \quad a_{14} \quad a_3 \quad a_9 \quad a_{15} \\ & a_2 \quad a_8 \quad a_{14} \quad a_4 \quad a_{10} \quad a_{16} \quad a_5 \quad a_{11} \quad a_{17} \\ & a_3 \quad a_9 \quad a_{15} \quad a_5 \quad a_{11} \quad a_{17} \quad a_6 \quad a_{12} \quad a_{18}]. \end{aligned}$$

19.2 玩家对称博弈的状态空间结构

记 n 人且 $|S_i|=\kappa$, $i=1,\cdots,n$ 之玩家对称博弈集合为 $\mathcal{S}_{[n;\kappa]}$. 由定理 19.1.1 可知, $\mathcal{S}_{[n;\kappa]}$ 是 $\mathcal{G}_{[n;\kappa]}$ 的一个线性子空间. 而且, 子空间的每个点由其首 κ^n 分量, 即 V_1^c, 唯一确定. (然后, 其余分量再由公式 (19.1.18) 算得.)

由例 19.1.1, 当 $n=2$ 时 V_1^c 可以任选, 而后 V_i^c, $i>1$ 由 V_1^c 唯一确定. 因此, 显见

$$\mathcal{S}_{[2,\kappa]} \sim \mathbb{R}^{\kappa^2}.$$

因此

$$\dim\left(\mathcal{S}_{[2,\kappa]}\right) = \kappa^2. \tag{19.2.1}$$

下面考虑 $n=3$. 利用 (19.1.18), V_1^c 满足

$$\left(V_1^c\right)^{\mathrm{T}} \in \ker\left(I_\kappa \otimes W_{[\kappa,\kappa]} - I_{\kappa^3}\right).$$

不难证明

$$\ker\left(I_\kappa \otimes W_{[\kappa,\kappa]} - I_{\kappa^3}\right) = I_\kappa \otimes \ker\left(W_{[\kappa,\kappa]} - I_{\kappa^2}\right).$$

所有的 $\{V_1^c\}$ 组成的子空间, 记作 $\mathcal{V}_1^c$, 则

$$\mathcal{V}_1^c = I_\kappa \otimes \ker\left(W_{[\kappa,\kappa]} - I_{\kappa^2}\right). \tag{19.2.2}$$

于是有如下结论:

命题 19.2.1 当 $n=3$,

$$\begin{aligned}\mathcal{V}_1^c = \operatorname{span}\Big\{&\delta_{\kappa^3}^{(i-1)\kappa+i+(t-1)\kappa^2}, \delta_{\kappa^3}^{(i-1)\kappa+j+(t-1)\kappa^2} \\ &+ \delta_{\kappa^3}^{(j-1)\kappa+i+(t-1)\kappa^2} \Big| i,t=1,\cdots,\kappa; i<j\leqslant\kappa\Big\}.\end{aligned} \tag{19.2.3}$$

证明 首先考虑

$$K_0 := \ker\left(W_{[\kappa,\kappa]} - I_{\kappa^2}\right).$$

记 $W_{[\kappa,\kappa]}$ 的元素为 $W_{[\kappa,\kappa]} = (w_{i,j})$. 那么, 在第 $(i-1)k+i$ 行有唯一的非零元素 $w_{(i-1)\kappa+i,(i-1)\kappa+i}=1$. 因此, 有

$$\delta_{\kappa^2}^{(i-1)\kappa+i} \in K_0, \quad i=1,\cdots,\kappa. \tag{19.2.4}$$

在 $(i-1)\kappa+j$, $j\neq i$ 行, 唯一的非零元素为 $w_{(i-1)\kappa+j,(j-1)\kappa+i}=1$. 因此, 有

$$\delta_{\kappa^2}^{(i-1)\kappa+j} + \delta_{\kappa^2}^{(j-1)\kappa+i} \in K_0, \quad i=1,\cdots,\kappa;\ j>i. \tag{19.2.5}$$

由于对称性, 为避免重复, 可以选择 $j>i$ 的向量或 $j<i$ 的向量. 因此, 形为 (19.2.4) 或 (19.2.5) 的向量形成 K_0 的一组基底.

由于 $\delta_{\kappa^2}^s \in K_0$ 当且仅当 $I_\kappa \otimes \delta_{\kappa^2}^s \in I_\kappa \otimes K_0$, 并且

$$I_\kappa \otimes \delta_{\kappa^2}^s = \left\{\delta_{\kappa^3}^{s+(t-1)\kappa^2} \,\middle|\, t=1,\cdots,\kappa\right\}.$$

结论显见. □

例 19.2.1　设 $n=3$.

1) 寻找 $\mathcal{S}_{[3,2]}$ 的基底. 利用命题 19.2.1, 该基底列表于表 19.2.2.

表 19.2.1　$\mathcal{S}_{[3,2]}$ 的基底

$t\backslash i(j)$	$i=1$	$i=2$	$i=1,\ j=2$
1	δ_8^1	δ_8^4	$\delta_8^2+\delta_8^3$
2	δ_8^5	δ_8^8	$\delta_8^6+\delta_8^7$

2) 寻找 $\mathcal{S}_{[3,3]}$ 的基底. 利用命题 19.2.1, 该基底列表于表 19.2.1.

表 19.2.2　$\mathcal{S}_{[3,3]}$ 的基底

$t\backslash i(j)$	$i=1$	$i=2$	$i=3$	$i=1,\ j=2$	$i=1,\ j=3$	$i=2,\ j=3$
1	δ_{27}^1	δ_{27}^5	δ_{27}^9	$\delta_{27}^2+\delta_{27}^4$	$\delta_{27}^3+\delta_{27}^7$	$\delta_{27}^6+\delta_{27}^8$
2	δ_{27}^{10}	δ_{27}^{14}	δ_{27}^{18}	$\delta_{27}^{11}+\delta_{27}^{13}$	$\delta_{27}^{12}+\delta_{27}^{16}$	$\delta_{27}^{15}+\delta_{27}^{17}$
3	δ_{27}^{19}	δ_{27}^{23}	δ_{27}^{27}	$\delta_{27}^{20}+\delta_{27}^{22}$	$\delta_{27}^{21}+\delta_{27}^{25}$	$\delta_{27}^{24}+\delta_{27}^{26}$

显然, 它们与例 19.1.1 的结果一致.

推论 19.2.1

$$\dim\left(\mathcal{S}_{[3,\kappa]}\right)=\frac{\kappa^2(\kappa+1)}{2}. \tag{19.2.6}$$

证明　显然

$$\dim\left(\mathcal{V}_1^c\right)=\kappa+\frac{\kappa(\kappa-1)}{2}=\frac{\kappa(\kappa+1)}{2}.$$

于是 $I_\kappa\otimes\mathcal{V}_1^c$ 维数为

$$\kappa\times\frac{\kappa(\kappa+1)}{2}=\frac{\kappa^2(\kappa+1)}{2}.$$

□

记

$$N_\kappa^n:=\binom{n+\kappa-1}{n}=\frac{(n+\kappa-1)!}{n!(\kappa-1)!}$$

为 k 项式 $(x_1+x_2+\cdots+x_k)^n$ 不同类的项的数目. 那么, 有如下引理:

引理 19.2.1　*考察 $G\in\mathcal{G}_{[n;\kappa]}$. 那么*

$$|\{\#s|s\in S\}|=N_\kappa^n. \tag{19.2.7}$$

证明　设 $x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_k^{\alpha_k}$ 为 $(x_1+x_2+\cdots+x_k)^n$ 展式的一项. 定义从 $(x_1+x_2+\cdots+x_k)^n$ 不同类项集合到 S 的一个映射为

$$\varphi: x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_\kappa^{\alpha_\kappa}\mapsto s\in S,$$

这里 $\#s = (\alpha_1, \alpha_2, \cdots, \alpha_k)$. 显然, 这是到 S 的等价类的一个一对一的映上, 这里, 每个等价类中的局势具有相同的策略指标. □

结合引理 19.2.1 与命题 19.2.1 可知

定理 19.2.1

$$\dim\left(\mathcal{S}_{[n;\kappa]}\right) = \kappa N_{\kappa}^{n-1}. \tag{19.2.8}$$

证明 我们只需考虑 $\mathcal{V}_1^c$. 首先注意玩家 1 具有 κ 个可能策略供选择. 而对应于每一个选择 S^{-1} 的每一个等价类形成一组基底. 结论显见. □

19.3 更名对称

考虑性别博弈, 其中有两个玩家, 玩家 P_1 (丈夫) 想去看足球赛, 而玩家 P_2 (妻子) 想去看歌剧. 支付函数见表 19.3.1.

表 19.3.1 性别战的支付双矩阵

$P_1\backslash P_2$	$s_2^1 = F$	$s_2^2 = O$
$s_1^1 = F$	$(4, 3)$	$(0,\ 0)$
$s_1^2 = O$	$(0,\ 0)$	$(3, 4)$

不难验证, 性别博弈不是玩家对称博弈. 但是, 如果将玩家 P_2 的两个策略重新更名一下, 即令 $s_2^1 = O$, $s_2^2 = F$, 那么, 显然性别博弈就成了玩家对称博弈. 受这个例子的启发, 我们给出以下定义:

定义 19.3.1([36]) *考查 $G \in \mathcal{G}_{[n;\kappa]}$. G 称为一个更名对称博弈, 如果存在一组映射 $\theta_i \in \mathbf{S}_\kappa$, $i = 1, \cdots, n$, 称为更名映射, 使得对任意 $x, y \in S$, 如果 $\theta_i(x_i) = \theta_j(y_j)$ 并且 $\#\theta_{-i}(x^{-i}) = \#\theta_{-j}(y^{-j})$, 那么*

$$c_i(\theta_i(x_i), \theta_{-i}(x^{-i})) = c_j(\theta_j(y_j), \theta_{-j}(y^{-j})). \tag{19.3.1}$$

比较 (19.3.1) 与 (19.1.5), 更名对称的意义是显而易见的.

令 $\eta \in \mathbf{S}_\kappa$ 及 $x = \delta_\kappa^i \in \Delta_\kappa$. 定义

$$M_\eta = \delta_\kappa[\eta(1), \eta(2), \cdots, \eta(\kappa)].$$

容易检验

$$M_\eta \delta_\kappa^i = \delta_\kappa^{\eta(i)}. \tag{19.3.2}$$

设 $(x_1, x_2, \cdots, x_n) \in S$ 为一个局势, 这里 $x_i \in S_i$, $i = 1, \cdots, n$. 在向量形式下将它表示为 $x = \ltimes_{i=1}^n x_i \in \Delta_{\kappa^n}$. 即, $S \sim \Delta_{\kappa^n}$. 设 $\Theta := (\theta_1, \theta_2, \cdots, \theta_n)$ 为一族更名映射, 并且

$$\Theta(x) = y := \ltimes_{i=1}^{n} \theta_i(x_i).$$

那么, 显然有

$$\begin{aligned} y &= \ltimes_{i=1}^{n} M_{\theta_i} x_i \\ &= [M_{\theta_1}(I_\kappa \otimes M_{\theta_2}) \cdots (I_{\kappa^{n-1}} \otimes M_{\theta_n})]\, x \\ &:= M_\Theta x. \end{aligned} \tag{19.3.3}$$

定理 19.3.1 $G \in \mathcal{G}_{[n;\kappa]}$ 为更名对称博弈, 当且仅当, 存在一组更名映射 $\theta := (\theta_1, \theta_2, \cdots, \theta_n)$, 使得

1)

$$V_1^c \left[M_\theta^{-1} \Phi_s M_\theta - I_{k^{s+1}} \right] = 0, \quad s = 2, 3, \cdots, n-1; \tag{19.3.4}$$

2)

$$V_i^c = V_1^c M_\theta^{-1} W_{[\kappa^{i-1}, \kappa]} M_\theta, \quad i = 2, 3, \cdots, n. \tag{19.3.5}$$

这里的 Φ_s 见 (19.1.17).

证明 令 $y = M_\theta x$ 且

$$c_1(x_1, \cdots, x_n) = V_1^x \ltimes_{i=1}^{n} x_i = V_1^y \ltimes_{i=1}^{n} y_i.$$

因为 $y = M_\theta x$, 有

$$V_1^x x = V_1^y y = V_1^y M_\theta x.$$

因此,

$$V_1^x = V_1^y M_\theta.$$

由于 G 相对于 y 是玩家对称博弈, 根据 (19.1.18) 可知

$$V_1^x x = V_1^y y = V_1^y \Phi_s y = V_1^x M_\theta^{-1} \Phi_s M_\theta x.$$

于是有

$$V_1^x = V_1^x M_\theta^{-1} \Phi_s M_\theta, \quad s = 2, \cdots, n-1.$$

这就证明了 (19.3.4).

(19.3.5) 类似可证. □

推论 19.3.1 $G \in \mathcal{G}_{[n;\kappa]}$ 是更名对称博弈且其更名映射矩阵为 M_{θ_i}, $i = 1, \cdots, n$, 当且仅当

$$V_i^c M_{\theta_i}^{\mathrm{T}}, \quad i = 1, \cdots, n$$

是一个玩家对称博弈.

证明 注意到 $M_{\theta_i}^{-1} = M_{\theta_i}^{\mathrm{T}}$, 结论直接来自 (19.3.4) 和 (19.3.5). □

例 19.3.1 考察 $G \in \mathcal{G}_{[3;3]}$, 它的支付函数如下:

$$
\begin{aligned}
V_1^c = [&a_1, a_2, a_3, a_4, a_5, a_2, a_6, a_4, a_1\\
&a_7, a_8, a_9, a_{10}, a_{11}, a_8, a_{12}, a_{10}, a_7\\
&a_{13}, a_{14}, a_{15}, a_{16}, a_{17}, a_{14}, a_{18}, a_{16}, a_{13}];\\
V_2^c = [&a_{12}, a_{10}, a_7, a_{18}, a_{16}, a_{13}, a_6, a_4, a_1\\
&a_7, a_8, a_9, a_{13}, a_{14}, a_{15}, a_1, a_2, a_3\\
&a_{10}, a_{11}, a_8, a_{16}, a_{17}, a_{14}, a_4, a_5, a_2];\\
V_3^c = [&a_1, a_{13}, a_7, a_4, a_{16}, a_{10}, a_6, a_{18}, a_{12}\\
&a_3, a_{15}, a_9, a_2, a_{11}, a_8, a_1, a_{13}, a_7\\
&a_2, a_{14}, a_8, a_5, a_{17}, a_{11}, a_4, a_{16}, a_{10}].
\end{aligned}
$$

令

$$
\theta = \{\theta_1 = (1,2,3),\ \theta_2 = (1,3,2),\ \theta_3 = (1,2)\}.
$$

那么, 容易算得

$$
\begin{aligned}
M_\theta = \delta_{27}[\ &17, 16, 18, 11, 10, 12, 14, 13, 15,\\
&26, 25, 27, 20, 19, 21, 23, 22, 24, 8, 7, 9, 2, 1, 3, 5, 4, 6].
\end{aligned}
$$

容易检验 (19.3.4) 及 (19.3.1) 均满足. 因此, G 是在更名 θ 下的更名对称博弈.

19.4 策略对称博弈

记所有玩家策略集的并:

$$
U := \cup_{i=1}^n S_i = \{1_1, 2_1, \cdots, \kappa_1; 1_2, 2_2, \cdots, \kappa_2; \cdots; 1_n, 2_n, \cdots, \kappa_n\},
$$

这里, $s_i \in S_i$, $i = 1, \cdots, n$.

定义 19.4.1 设 $\sigma \in \mathbf{S}_{n\kappa}$ 为 U 上的一个置换. σ 称为玩家导向的置换, 如果它满足: 如果存在 s_i 使得 $\sigma(s_i) = r_j \in S_j$, 那么, $\sigma(q_i) \in S_j$, $q = 1, \cdots, \kappa$. 玩家导向的置换集记作 $\mathcal{P}_{n;\kappa}$.

以下的结论直接来自定义:

命题 19.4.1 1) $\mathcal{P}_{n;\kappa}$ 是 $\mathbf{S}_{n\kappa}$ 的子群, 即,

$$
\mathcal{P}_{n;\kappa} < \mathbf{S}_{n\kappa}.
$$

2) $\mathcal{P}_{n;\kappa}$ 的元素个数为

$$|\mathcal{P}_{n;\kappa}| = n!(\kappa!)^n. \tag{19.4.1}$$

3) $\sigma \in \mathcal{P}_{n\kappa}$ 可以唯一确定一个 $\pi_\sigma \in \mathbf{S}_n$, 使得

$$\sigma(s_i) \in S_{\pi_\sigma(i)}, \quad \forall s_i \in S_i.$$

定义 19.4.2 ([115]) $G \in \mathcal{G}_{[n;\kappa]}$ 称为相对于 $\theta \in \mathcal{P}_{n;\kappa}$ 的策略对称博弈, 如果

$$c_i(s_1, \cdots, s_n) = c_{\pi_\theta(i)}\left(\theta(s_1), \cdots, \theta(s_n)\right), \quad s_i \in S_i;\ i = 1, \cdots, n. \tag{19.4.2}$$

设 $G \in \mathcal{G}_{[n;\kappa]}$. 所有的 $\theta \in \mathcal{P}_{n;\kappa}$ 满足 (19.4.2) 记作 $\Theta(G)$. 设 $\theta \in \Theta(G)$, 则它可导出 $\pi_\theta \in \mathbf{S}_n$.

$$\Pi(G) := \{\pi_\theta | \theta \in \Theta(G)\}.$$

下面这个结果是显然的[119]:

命题 19.4.2 设 $G \in \mathcal{G}_{[n;\kappa]}$. 那么,

1)

$$\Theta(G) < \mathcal{P}_{n\kappa};$$

2)

$$\Pi(G) < \mathbf{S}_n.$$

类似上一节的讨论可知

定理 19.4.1 设 $G \in \mathcal{G}_{[n;\kappa]}$ 及 $\theta \in \mathcal{P}_{n;\kappa}$. $\theta \in \Theta(G)$, 当且仅当

$$V_i^c = V_{\pi_\theta(i)}^c M_{\pi_\theta}\left[M_{\theta_1}\left(I_\kappa \otimes M_{\theta_2}\right) \cdots \left(I_{\kappa^{n-1}} \otimes M_{\theta_n}\right)\right]. \tag{19.4.3}$$

下面讨论几个例子.

例 19.4.1 设 $G \in \mathcal{G}_{[3;3]}$.

1) $\Theta(G) = \langle\theta\rangle$, 这里

$$\theta := \begin{array}{ccccccccc} 1_1 & 2_1 & 3_1 & 1_2 & 2_2 & 3_2 & 1_3 & 2_3 & 3_3 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2_3 & 3_3 & 1_3 & 3_2 & 1_2 & 2_2 & 2_1 & 1_1 & 3_1 \end{array}$$

显然

$$\pi_\theta = (1, 3).$$

于是有

$$M_{\pi_\theta} = W_{[3,3]} W_{[3,9]}.$$

$$M_\theta := M_{\theta_1}\left(I_3 \otimes M_{\theta_2}\right)\left(I_9 \otimes M_{\theta_3}\right),$$

它与例 19.3.1 中的 M_θ 一样. 下面考察 V_1^c, 因为 $V_3^c = V_1^c M_{\pi_\theta} M_\theta$ 以及 $V_1^c = V_3^c M_{\pi_\theta} M_\theta$, 有

$$V_1^c = V_1^c M_{\pi_\theta} M_\theta M_{\pi_\theta} M_\theta. \tag{19.4.4}$$

为满足 (19.4.4), V_1^c 应为

$$\begin{aligned} V_1^c = [&a_1, a_2, a_1, a_1, a_2, a_1, a_1, a_2, a_1,\\ &a_3, a_4, a_5, a_3, a_4, a_5, a_3, a_4, a_5,\\ &a_5, a_4, a_3, a_5, a_4, a_3, a_5, a_4, a_3]. \end{aligned}$$

于是

$$\begin{aligned} V_3^c &= V_1^c M_{\pi_\theta} M_\theta\\ &= [a_4, a_2, a_4, a_4, a_2, a_4, a_4, a_2, a_4,\\ &\quad a_5, a_1, a_3, a_5, a_1, a_3, a_5, a_1, a_3,\\ &\quad a_3, a_1, a_5, a_3, a_1, a_5, a_3, a_1, a_5]. \end{aligned}$$

下面考虑 V_2^c, 因为

$$V_2^c = V_2^c M_{\pi_\theta} M_\theta,$$

有

$$\begin{aligned} V_2^c = [&b_1, b_2, b_1, b_1, b_2, b_1, b_1, b_2, b_1\\ &b_3, b_1, b_3, b_3, b_1, b_3, b_3, b_1, b_3\\ &b_3, b_1, b_3, b_3, b_3, b_3, b_3, b_1, b_3]. \end{aligned}$$

2) 设 $\Theta(G) \supset \langle \theta, \eta \rangle$, 这里 θ 同上,

$$\eta := \begin{array}{ccccccccc} 1_1 & 2_1 & 3_1 & 1_2 & 2_2 & 3_2 & 1_3 & 2_3 & 3_3\\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow\\ 3_2 & 1_2 & 2_2 & 2_1 & 3_1 & 1_1 & 3_3 & 2_3 & 1_3 \end{array}$$

类似的计算可知, G 的支付函数满足: $V_1^c = V_2^c = V_3^c = \mu \times \mathbf{1}_{27}$, 这里 $\mu \in \mathbb{R}$. 这是一个平凡博弈, $\Pi(G) = S_3$ 并且 $\Theta(G) = \mathcal{P}_{3;3}$.

数值计算显示

$$|\langle \theta, \eta \rangle| = 648.$$

利用 (19.4.1), 有

$$|\Theta(G)| = |\mathcal{P}_{3;3}| = 1296.$$

因此, 不存在这样的 G, 它满足 $\Theta(G) = \langle \theta, \eta \rangle$.

最后, 不难看出, 玩家对称博弈和更名对称博弈都是策略对称博弈的特殊情况. 显然, 玩家对称博弈是更名对称博弈的特殊形式, 此时更名映射为恒等映射. 考虑更名对称博弈与策略对称博弈, 我们有:

命题 19.4.3 设 $G \in \mathcal{G}_{[n;\kappa]}$. G 是更名对称博弈, 当且仅当 G 是策略对称博弈, 且满足

$$\Pi(G) = \mathbf{S}_n; \tag{19.4.5}$$

并且

$$\Theta(G) \simeq \mathbf{S}_n. \tag{19.4.6}$$

证明 设 G 为一更名对称博弈. 定义一个映射 $\varphi : \mathbf{S}_n \to \Theta(G)$ 为 $\tau \mapsto \theta_\tau$, 这里

$$\theta_\tau : \begin{array}{cccc} 1_i & 2_i & \cdots & \kappa_i \\ \downarrow & \downarrow & \cdots & \downarrow \\ r_i(1)_{\tau(i)} & r_i(2)_{\tau(i)} & \cdots & r_i(\kappa)_{\tau(i)} \end{array}, \quad i = 1, \cdots, n,$$

其中 $r_1, \cdots, r_n$ 为更名映射集.

直接检验即可得到结论. □

19.5 注释与参考

本章讨论了三种对称博弈: 玩家对称博弈、更名对称博弈及策略对称博弈. 讨论的重点是它们的定义、状态空间结构和相互关系. 对称博弈虽然在自然界和社会现象中大量存在, 但目前研究得还很不够. 特别是关于它们的更深刻的性质, 目前知道的还不是很多.

第20章 合作博弈

20.1 特征函数

定义 20.1.1 一个合作博弈可以用一个二元结构 (N, v) 来描述. 其中 $N = \{1, \cdots, n\}$ 为局中人; $v : 2^N \to \mathbb{R}$ 是一个集函数, 满足 $v(\varnothing) = 0$, 称为特征函数.

v 刻画一个 n 人合作对策. 设 $S \in 2^N$, 也就是说 $S \subset N$, 它称为一个 (由 S 中成员结成的) 联盟. 那么, $v(S)$ 就表示在这个对策下 S 这个联盟的收益.

设 $S \in 2^N$, 可以用 $\mathcal{D}^n$ 来表示它. 记 $I_S = (s_1, \cdots, s_n) \in \mathcal{D}^n$, 这里

$$
s_j = \begin{cases} 1, & j \in S, \\ 0, & j \notin S. \end{cases}
$$

那么, 每一个特征函数就可以看作一个伪布尔函数

$$
v(S) = v(s_1, \cdots, s_n). \tag{20.1.1}
$$

然后, 令 $1 \sim \delta_2^1$, $0 \sim \delta_2^2$, 则有 $s_j \in \Delta_2$, $j = 1, 2, \cdots, n$. 那么, 对于每一个特征函数 v, 都可以找到它的结构向量, 记作 V_v, 使得

$$
v(S) = V_v \ltimes_{i=1}^n s_i. \tag{20.1.2}
$$

注意到 $V_v \in \mathbb{R}^{2^n}$, 并且, 由 $v(\varnothing) = 0$, V_v 的最后一个分量为 0. 由此可知

命题 20.1.1 令 $|N| = n$. 则 N 上所有对策, 记为 $G(N)$, 形成一个 $2^n - 1$ 维线性空间. 实际上, 它同构于 $\mathbb{R}^{2^n - 1}$.

定义 20.1.2 1) 称一个对策 v 满足超可加性 (super-additivity), 如果对任何两个联盟 P, $Q \in 2^N$ 且 $P \cap Q = \varnothing$ 均满足

$$
v(P \cup Q) \geqslant v(P) + v(Q). \tag{20.1.3}
$$

2) 如果对任何两个联盟 P, $Q \in 2^N$ 且 $P \cap Q = \varnothing$ 均满足

$$
v(P \cup Q) = v(P) + v(Q), \tag{20.1.4}
$$

则 v 称为非本质对策.

合作博弈通常期望合作有利可图, 因此, 通常只对满足超可加性的对策有兴趣. 同样道理, 非本质对策虽然也满足超可加性, 但由于合作不能使总收益增加, 故失去了合作的基础.

定理 20.1.1([9])　合作对策 v 为非本质对策的充要条件是

$$v(N)=\sum_{i=1}^{n} v(\{i\}). \tag{20.1.5}$$

考虑到超可加性, 将满足

$$v(N)>\sum_{i=1}^{n} v(\{i\})$$

的对策称为本质对策.

下面给几个简单例子.

例 20.1.1　设有 $N=\{1,2,\cdots,n\}$, 每人有一只手套.

- $L \subset N$: $i \in L$ 表示玩家 i 有一只左手套.
- $R \subset N$: $i \in R$ 表示玩家 i 有一只右手套.

假定一双手套价值 3 元, 而一只手套价值 0.05 元. 令 $S \in 2^N$, 则 S 的价值为

$$v(S)=3\times\min\left(|S\cap L|,|S\cap R|\right)+0.05\times\left(|S|-2\times\min(|S\cap L|,|S\cap R|)\right).$$

不难验证 v 为本质对策.

例 20.1.2　买卖马问题: 卖主 (玩家 1) 有马要卖, 马对他没有用处, 不卖则价值为零. 买主 (玩家 2) 愿出至多 1000 元, 买主 (玩家 3) 愿出至多 1100 元. 将该问题看作合作博弈, 求特征函数.

假如 1 将马以 x 元卖给 2, 1 可得 x 元, 2 可得 $1000-x$ 元. 显然, 1 与 2 合作的价值为

$$v(\{1,2\})=1000.$$

同理

$$v(\{1,3\})=1100.$$

如果 3 人同时在场, 1 必然会将马卖给 3, 因此

$$v(\{1,2,3\})=1100.$$

此外, 2, 3 结盟, 价值为零. 单人的价值也是零. 如果用结构向量表示, 则有

$$V_v=[1100,1000,1100,0,0,0,0,0].$$

20.2 常和博弈的特征函数

考虑二人零和博弈, 这时, 局中人的利益是对立的, 你赢则我输, 因此, 没有合作的基础. 二人常和博弈在本质上与二人零和博弈是一样的. 因此, 我们考虑至少 3 个人的情况. 这时, 即使是零和博弈, 合作还是有意义的. 先看下面这个例子.

例 20.2.1 三个人玩手心手背. 如果三人出手一样, 则无输赢. 否则, 落单者向其他二个每人支付 1 元. 于是, 其支付矩阵见表 20.2.1.

表 20.2.1 例 20.2.1 支付函数

支付/局势	111	112	121	122	211	212	221	222
c_1	0	1	1	−2	−2	1	1	0
c_2	0	1	−2	1	1	−2	1	0
c_3	0	−2	1	1	1	1	−2	0

我们将最佳收益作为特征函数. 由于是零和, 有: $v(1,2,3)=0$. 考虑 $v(1,2)$. 如果将 $R=\{1,2\}$ 作为一方, $R^c=\{3\}$ 作为一方. 那么, R 的收益矩阵可如下表示 (表 20.2.2).

表 20.2.2 R 对 R^c 收益函数

$R=\{1,2\}\backslash\ R^c=\{3\}$	1	2
11	0	2
12	−1	−1
21	−1	−1
22	2	0

不管 3 选什么策略, R 选 12 或 21 都劣于 11 或 22. 所以 R 不可能选它们, 因此, 可删去第 2 行和第 3 行. 因此, R 和 R^c 各有两个策略: 记 $p=P(R=11)$, $q=P(R^c=1)$ 则 R 的期望值为

$$E_R=p(1-q)\times 2+(1-p)q\times 2.$$

最优策略为 $11\left(\frac{1}{2}\right)+22\left(\frac{1}{2}\right)$. 此时 $E_R=1$. 同时, R^c 的最优策略为 $1\left(\frac{1}{2}\right)+2\left(\frac{1}{2}\right)$, 此时 $E_{R^c}=-1$. 于是, 定义

$$v(\{1,2\})=1,\quad v(\{3\})=-1.$$

由对称性容易得出, v 的结构向量为

$$V_v=[0,1,1,-1,1,-1,-1,0].$$

下面考虑一般的 n 人常和 (含零和) 博弈, 它满足

$$v(N) := \sum_{i=1}^{n} c_i(s) = \text{const.}, \quad \forall s \in S.$$

现在考虑 $\varnothing \neq R \subsetneq N$. 其余集 $R^c \neq \varnothing$. 要考虑联盟 R 的价值, 很自然地定义为它在与 R^c 博弈中的所得. 记 R 及 R^c 的策略分别为

$$S_R = \prod_{i \in R} S_i, \quad S_{R^c} = \prod_{i \in R^c} S_i.$$

R 与 R^c 之间的博弈变为二人常和博弈. 于是, 可将均衡值定义为 R 的特征值. 根据矩阵博弈的基本性质 (参见第 15 章), 有

$$\begin{aligned} v(R) &:= \max_{\xi \in \bar{S}_R} \min_{\eta \in \bar{S}_{R^c}} \sum_{r \in R} e_r(\xi, \eta) \\ &= \min_{\eta \in \bar{S}_{R^c}} \max_{\xi \in \bar{S}_R} \sum_{r \in R} e_r(\xi, \eta). \end{aligned} \tag{20.2.1}$$

由 (20.2.1) 所定义的特征函数称为常和博弈特征函数.

显然, 例 20.2.1 中的特征函数就是按 (20.2.1) 定义的常和博弈特征函数.

下面讨论常和博弈特征函数的一些性质:

命题 20.2.1　设 v 为常和博弈特征函数. 则

$$v(R) + v(R^c) = v(N), \quad \forall R \in 2^N. \tag{20.2.2}$$

证明　考虑 $\varnothing \neq R \subsetneq N$.

$$\begin{aligned} v(R) &= \max_{\xi \in \bar{S}_R} \min_{\eta \in \bar{S}_{R^c}} \sum_{r \in R} e_r(\xi, \eta) \\ &= \max_{\xi \in \bar{S}_R} \min_{\eta \in \bar{S}_{R^c}} \left(v(N) - \sum_{r \in R^c} e_r(\xi, \eta) \right) \\ &= v(N) - \min_{\xi \in \bar{S}_R} \max_{\eta \in \bar{S}_{R^c}} \sum_{r \in R^c} e_r(\xi, \eta) \\ &= v(N) - v(R^c). \end{aligned}$$

□

命题 20.2.2　设 v 为常和博弈特征函数. 则 v 具有超可加性. 即, 设 $R, T \in 2^N$, $R \cap T = \varnothing$, 则

$$v(R) + v(T) \leqslant v(R \cup T). \tag{20.2.3}$$

证明 定义 $e_R(\xi,\eta) := \sum_{r\in R} e_r(\xi,\eta)$. 那么, 有

$$
\begin{aligned}
v(R\cup T) &= \min_{\eta\in \overline{R\cup T}^c} \max_{\xi\in \overline{R\cup T}} e_{R\cup T}(\xi,\eta)\\
&= \min_{\eta\in \overline{R\cup T}^c} \max_{\alpha\in \bar{R}} \max_{\beta\in \bar{T}} (e_R(\alpha,\beta,\eta)+e_T(\alpha,\beta,\eta))\\
&\geqslant \min_{\eta\in \overline{R\cup T}^c} \max_{\alpha\in \bar{R}} \left(\min_{\beta\in \bar{T}} e_R(\alpha,\beta,\eta)+\max_{\beta\in \bar{T}} e_T(\alpha,\beta,\eta)\right)\\
&\geqslant \min_{\eta\in \overline{R\cup T}^c} \left(\max_{\alpha\in \bar{R}} \min_{\beta\in \bar{T}} e_R(\alpha,\beta,\eta)+\min_{\alpha\in \bar{R}} \max_{\beta\in \bar{T}} e_T(\alpha,\beta,\eta)\right)\\
&\geqslant \min_{\eta\in \overline{R\cup T}^c} \max_{\alpha\in \bar{R}} \min_{\beta\in \bar{T}} e_R(\alpha,\beta,\eta)+\min_{\eta\in \overline{R\cup T}^c} \min_{\alpha\in \bar{R}} \max_{\beta\in \bar{T}} e_T(\alpha,\beta,\eta)\\
&= \min_{\xi\in \bar{R}^c} \max_{\alpha\in \bar{R}} e_R(\alpha,\xi)+\min_{\xi\in \bar{T}^c} \max_{\beta\in \bar{T}} e_T(\beta,\xi)\\
&= v(R)+v(T).
\end{aligned}
$$

□

对于非常和博弈, 我们能否利用

$$
v(R) := \max_{\xi\in \bar{S}_R} \min_{\eta\in \bar{S}_{R^c}} e_R(\xi,\eta), \tag{20.2.4}
$$

及

$$
v(N) = \max_{s\in S} \sum_{i\in N} c_i(s)
$$

来定义特征函数 v 呢? 从物理意义上说, 这未必合理, 因你的极大点未必是对方的极小点. 因此, 这个 v 值可能是保守的. 从理论上看, 它可能不满足超可加性.

20.3 两种特殊的博弈

20.3.1 无异议博弈

定义 20.3.1 一个合作博弈 (N,v) 称为无异议博弈 (unanimity game), 如果存在一个 $\varnothing \neq T \in 2^N$, 使得

$$
v(S) := u_T(S) = \begin{cases} 1, & T\subset S,\\ 0, & \text{其他}. \end{cases} \tag{20.3.1}
$$

无异议博弈的特征函数, 简称无异议特征函数, 构成特征函数空间的基底.

定理 20.3.1　1) 设 $N=\{1,2,\cdots,n\}$. 其所有特征函数集合, 记作 G^N, 是一个 2^n-1 维线性空间.

2) 无异议特征函数集合

$$\left\{u_T \middle| \varnothing \neq T \in 2^N\right\},$$

构成 G^N 的一个基底. 并且可知: 令 $v\in G^N$. 则

$$v=\sum_{T\in 2^N\backslash\varnothing}\mu_T u_T, \tag{20.3.2}$$

这里

$$\mu_T=\sum_{S\subset T}(-1)^{(|T|-|S|)}v(S). \tag{20.3.3}$$

证明　对任意的 $R\in 2^N$, 有

$$\begin{aligned}
\left(\sum_{T\in 2^N\backslash\varnothing}\mu_T u_T\right)(R) &= \sum_{T\in 2^N\backslash\varnothing}\mu_T u_T(R)\\
&=\sum_{\varnothing\neq T\subset R}\mu_T\\
&=\sum_{\varnothing\neq T\subset R}\sum_{S\subset T}(-1)^{(|T|-|S|)}v(S)\\
&=\sum_{S\subset R}\left(\sum_{S\subset T\subset R}(-1)^{(|T|-|S|)}\right)v(S)\\
&=v(R)+\sum_{\substack{S\subset R\\ S\neq R}}\sum_{t=|S|}^{|R|}(-1)^{t-|S|}\binom{|R|-|S|}{t-|S|}v(S)\\
&=v(R).
\end{aligned}$$

最后一个等式来自二项式定理, 即对正整数 $r>0$, 有

$$\sum_{i=0}^{r}(-1)^i\binom{r}{i}=(1-1)^r=0.$$

因此, 对 $|S|<|R|$ 有

$$\sum_{t=|S|}^{|R|}(-1)^{t-|S|}\binom{|R|-|S|}{t-|S|}=0.$$

即

$$\left(\sum_{T\in 2^N\backslash\varnothing}\mu_T u_T\right)(R)=v(R),\quad \forall R\in 2^N.$$

于是, 立刻得到 (20.3.2)–(20.3.3).　□

例 20.3.1 令 $N=\{1,2\}$.

$$S_1=1\,1,\quad S_2=1\,0,\quad S_3=0\,1,\quad S_4=0\,0.$$

根据 (20.3.1), 有

$$\begin{aligned}
&u_{S_1}(S_1)=1,\quad u_{S_1}(S_2)=0,\quad u_{S_1}(S_3)=0,\quad u_{S_1}(S_4)=0,\\
&u_{S_2}(S_1)=1,\quad u_{S_2}(S_2)=1,\quad u_{S_2}(S_3)=0,\quad u_{S_2}(S_4)=0,\\
&u_{S_3}(S_1)=1,\quad u_{S_3}(S_2)=0,\quad u_{S_3}(S_3)=1,\quad u_{S_3}(S_4)=0,
\end{aligned}$$

利用 (20.3.2), 有

$$v=\mu_{S_1}u_{S_1}+\mu_{S_2}u_{S_2}+\mu_{S_3}u_{S_3},$$

这里 μ_{S_i} 可由 (20.3.3) 计算如下:

$$\begin{aligned}
\mu_{S_1}&=\sum_{S\subset S_1}(-1)^{(|S_1|-|S|)}v(S)=v(S_1)-v(S_2)-v(S_3);\\
\mu_{S_2}&=\sum_{S\subset S_2}(-1)^{(|S_2|-|S|)}v(S)=v(S_2);\\
\mu_{S_3}&=\sum_{S\subset S_3}(-1)^{(|S_3|-|S|)}v(S)=v(S_3).
\end{aligned}$$

因此可得

$$v=[v(S_1)-v(S_2)-v(S_3)]\,u_{S_1}+v(S_2)u_{S_2}+v(S_3)u_{S_3}. \tag{20.3.4}$$

虽然, 这个基底有许多应用[32], 但从例 20.3.1 不难看出, 经典的系数计算公式 (20.3.3) 用起来很不方便. 下面给出一套简单的计算公式. 为方便计, 我们形式地引入空集的无异议特征函数:

$$u_{\varnothing}(S):=\begin{cases}1, & S=\varnothing,\\ 0, & \text{其他}.\end{cases}$$

同时约定

$$\mu_{\varnothing}=0.$$

那么, 式 (20.3.2) 可形式地改写成

$$v=\sum_{T\in 2^N}\mu_T u_T. \tag{20.3.5}$$

下面将 u_T 值用列表的形式表示出来, 用结构向量 V_T, V_S 来表示不同的 T 和 S. 当 $|N|=1$, $|N|=2$ 和 $|N|=3$ 时, u_T 的值分别用表 20.3.1, 20.3.2, 20.3.3 来表示.

表 20.3.1 $|N|=1$ 时的 u_T

$V_T \backslash u_T \backslash V_S$	1	0
1	1	0
0	1	1

表 20.3.2 $|N|=2$ 时的 u_T

$V_T \backslash u_T \backslash V_S$	1 1	1 0	0 1	0 0
1 1	1	0	0	0
1 0	1	1	0	0
0 1	1	0	1	0
0 0	1	1	1	1

表 20.3.3 $|N|=3$ 时的 u_T

$V_T \backslash u_T \backslash V_S$	1 1 1	1 1 0	1 0 1	1 0 0	0 1 1	0 1 0	0 0 1	0 0 0
1 1 1	1	0	0	0	0	0	0	0
1 1 0	1	1	0	0	0	0	0	0
1 0 1	1	0	1	0	0	0	0	0
1 0 0	1	1	1	1	0	0	0	0
0 1 1	1	0	0	0	1	0	0	0
0 1 0	1	1	0	0	1	1	0	0
0 0 1	1	0	1	0	1	0	1	0
0 0 0	1	1	1	1	1	1	1	1

将表中的 u_T 值作成矩阵, 记为 U_n, 称为 n 阶无异议矩阵, 这里 $n=|N|$, $U_u \in \mathcal{B}_{2^n \times 2^n}$.

由表 20.3.1–20.3.3, 不难发现如下的结构特点.

命题 20.3.1 无异议矩阵可递推地构造如下:

$$
\begin{cases}
U_1 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \\
U_{k+1} = \begin{bmatrix} U_k & 0 \\ U_k & U_k \end{bmatrix}, \quad k=2,3,\cdots.
\end{cases}
\tag{20.3.6}
$$

证明 将 U_{k+1} 均分为四块:

$$U_{k+1}=\begin{bmatrix}U_{11} & U_{12}\\ U_{21} & U_{22}\end{bmatrix}.$$

考虑 U_{11}. 记 T_i 和 S_j 分别为其第 i 行所对应的 T 值和第 j 列所对应的 S 值. 相应地, 记 T_i' 和 S_j' 分别为 U_k 第 i 行所对应的 T 值和第 j 列所对应的 S 值. 不难发现 $V_{T_i}=1\times V_{T_i'}$ 及 $V_{S_i}=1\times V_{S_i'}$. (即 V_{T_i} 可由 $V_{T_i'}$ 在前面加 1 而得到, 等.) 因此, T_i 和 S_i 的包含关系与 T_i' 和 S_i' 的包含关系一样. 于是有 $U_{11}=U_k$. 同样可知 $U_{21}=U_{22}=U_k$. 至于 U_{12}, 它对应的 T_i 包含第一个元素, 而 S_j 不包含第一个元素, 故 $T_i\not\subset S_j, \forall i,j$. 因此, $U_{12}=0$. □

现在, 令 $v\in G^N$, 记其结构向量为

$$V_v=(v_1\ v_2\ \cdots\ v_{2^n}).$$

设它有如下展开式

$$v=\sum_{i=1}^{2^n}\mu_i u_{T_i}, \tag{20.3.7}$$

这里 $\mu_{2^n}=0$ 是固定的. 于是有

定理 20.3.2 特征函数 v 的结构向量满足

$$V_v=(\mu_1\ \mu_2\ \cdots\ \mu_{2^n})U_n. \tag{20.3.8}$$

因此, 展式 (20.3.7) 的系数满足

$$(\mu_1\ \mu_2\ \cdots\ \mu_{2^n})=V_vU_n^{-1}. \tag{20.3.9}$$

注意, 在 (20.3.8) 或 (20.3.9) 中, $v_{2^n}=0$ 与 $\mu_{2^n}=0$ 互相对应.

不难看出, U_n^{-1} 可由下式得到

$$\begin{cases}U_1^{-1} &= \begin{bmatrix}1 & 0\\ -1 & 1\end{bmatrix},\\ U_{k+1}^{-1} &= \begin{bmatrix}U_k^{-1} & 0\\ -U_k^{-1} & U_k^{-1}\end{bmatrix}, \quad k=2,3,\cdots.\end{cases} \tag{20.3.10}$$

为检验定理 20.3.2, 再与例 20.3.1 比较一下.

例 20.3.2 回忆例 20.3.1. 当 $n=2$ 时, 利用 (20.3.8) 有

$$(v(S_1), v(S_2), v(S_3), 0) = (\mu_1, \mu_2, \mu_3, \mu_4)\, U_2.$$

因此

$$\begin{aligned}(\mu_1, \mu_2, \mu_3, \mu_4) &= (v(S_1), v(S_2), v(S_3), 0)\, U_2^{-1} \\ &= (v(S_1), v(S_2), v(S_3), 0) \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 1 \end{bmatrix} \\ &= (v(S_1) - v(S_2) - v(S_3), v(S_2), v(S_3), 0)\,.\end{aligned}$$

这就是 (20.3.4).

20.3.2　规范博弈

前面介绍过超可加性, 它是合作的基础. 本小节只讨论满足超可加性的合作博弈.

定义 20.3.2　设 (N, v) 和 (N, v') 为两个合作博弈. 称特征函数 v 和 v' 为策略等价的, 记作 $v \sim v'$, 如果存在 $\alpha > 0$, $\beta_i \in \mathbb{R}$, $i = 1, \cdots, n(n = |N|)$, 使得

$$v'(R) = \alpha v(R) + \sum_{i \in R} \beta_i, \quad \forall R \in 2^N. \tag{20.3.11}$$

容易验证, 如果 v 满足超可加性, $v \sim v'$, 则 v' 也满足超可加性.

定义 20.3.3　合作博弈 (N, v) 称为一个 $(0,1)$ 规范博弈 (normalization game), 如果它满足

(i)

$$v(\{i\}) = 0, \quad \forall i \in N;$$

(ii)

$$v(N) = 1.$$

引理 20.3.1　(满足超可加性的) 合作博弈 (N, v) 是非本质合作博弈的充要条件是

$$v(N) = \sum_{i=1}^{n} v(\{i\}). \tag{20.3.12}$$

证明　(必要性) 反复使用 (20.1.4) 即得.

(充分性) 设 $R, T \in 2^N$, $R \cap T = \varnothing$.

$$\begin{aligned}
v(N) &= \sum_{i=1}^{n} v(\{i\}) \\
&= \sum_{i \in R} v(\{i\}) + \sum_{i \in T} v(\{i\}) + \sum_{i \in (R \cup T)^c} v(\{i\}) \\
&\leqslant v(R) + v(T) + v((R \cup T)^c) \\
&\leqslant v(R \cup T) + v((R \cup T)^c) \\
&\leqslant v(N).
\end{aligned}$$

于是可得

$$v(R) + v(T) = v(R \cup T).$$

□

推论 20.3.1 (满足超可加性的) 合作博弈 (N, v) 是本质合作博弈的充要条件是

$$v(N) > \sum_{i=1}^{n} v(\{i\}). \tag{20.3.13}$$

定理 20.3.3 每一个本质合作博弈 (N, v) 都与唯一的一个 $(0,1)$ 规范博弈策略等价.

证明 (存在性) 因为是本质博弈, 则

$$v(N) - \sum_{i=1}^{n} v(\{i\}) > 0.$$

令

$$\alpha = \frac{1}{v(N) - \sum\limits_{i=1}^{n} v(\{i\})} > 0;$$

$$\beta_i = -\alpha v(\{i\}), \quad i = 1, \cdots, n.$$

定义

$$v'(R) = \alpha v(R) + \sum_{i \in R} \beta_i, \quad \forall R \in 2^N.$$

容易验证, v' 是 $(0,1)$ 规范博弈.

(唯一性) 设 $v'' \sim v$ 是另一个 $(0,1)$ 规范博弈, 则 $v'' \sim v'$. 于是, 存在 $\alpha' > 0$, β_i', $i = 1, \cdots, n$, 使得

$$v''(R) = \alpha' v'(R) + \sum_{i \in R} \beta_i', \quad \forall R \in 2^N. \tag{20.3.14}$$

可知

$$\begin{aligned} v''(\{i\}) &= \alpha' v'(\{i\}) + \beta_i', \quad \forall i \in N, \\ v''(N) &= \alpha' v'(N) + \sum_{i=1}^{n} \beta_i'. \end{aligned}$$

考虑到

$$\begin{aligned} v''(\{i\}) &= v'(\{i\}) = 0, \quad \forall i \in N, \\ v''(N) &= \alpha' v'(N) = 1, \end{aligned}$$

即得

$$\beta_i' = 0, \quad \forall i \in N; \qquad \alpha' = 1.$$

因此, $v'' = v'$. □

对于非本质合作博弈 (N, v), 有

$$v(R) = \sum_{i \in R} v(\{i\}), \quad \forall R \in 2^N.$$

令 $\alpha = 1$, $\beta_i = -v(\{i\})$, 定义

$$v'(R) = v(R) - \sum_{i \in R} v(\{i\}).$$

显然有 $v'(R) = 0$, $\forall R \in 2^N$. 称 v' 为零规范博弈. 则得

命题 20.3.2 每一个非本质合作博弈都与零规范博弈策略等价.

20.4 分 配

分配是合作博弈研究的中心问题, 人们寻找的是一个合理的利益分配, 使每个理性参与者都满意.

定义 20.4.1 给定一个合作博弈 (N, v), 一个 n 维向量 $x = (x_1, \cdots, x_n)$ 称为一个分配, 如果它满足

(i) 个体合理性 (individual rationality)

$$x_i \geqslant v(\{i\}), \quad i = 1, \cdots, n;$$

(ii) **群体合理性** (group rationality)

$$\sum_{i=1}^{n} x_i = v(N).$$

(N,v) 的分配集记作 $E(v)$.

注意, 个体合理性保证每个人合作所得不少于单干. 否则, 个体会拒绝合作; 群体合理性保证合作的收益既被分光, 又不至于出现入不敷出的 "空头支票".

命题 20.4.1 *非本质博弈只有一个分配, 即*

$$x_i = v(\{i\}), \quad i = 1, \cdots, n. \tag{20.4.1}$$

证明 容易检验, 式 (20.4.1) 定义的确实是一个分配. 假设存在另一个分配 $x' \neq x$, 那么, 至少有一个 $x'_i > v(\{i\})$. 于是

$$v(N) = \sum_{i=1}^{n} x'_i > \sum_{i=1}^{n} v(\{i\}) = v(N).$$

矛盾. 故 $x' = x$. □

命题 20.4.2 *本质博弈的分配构成一个 n 维非空闭凸集.*

证明 设 (N,v) 为一本质博弈. 则

$$v(N) - \sum_{i=1}^{n} v(\{i\}) := d > 0.$$

设 $\alpha_i \geqslant 0$, 且 $\sum_{i=1}^{n} \alpha_i = d$. 则不难看出: 任何一个分配 x 均可表示为

$$x_i = v(\{i\}) + \alpha_i, \quad i = 1, \cdots, n. \tag{20.4.2}$$

定义一组分配 z^i, $i = 1, \cdots, n$ 如下:

$$z^i_j := \begin{cases} v(\{j\}), & j \neq i, \\ v(\{j\}) + d, & j = i. \end{cases}$$

则由式 (20.4.2) 定义的分配可表示为

$$x = \sum_{i=1}^{n} \left(\frac{\alpha_i}{d}\right) z^i.$$

记 $z^0 = (v(\{1\}), v(\{2\}), \cdots, v(\{n\}))$. 由于 $\{z^i - z^0 \mid i = 1, \cdots, n\}$ 线性无关, $\{z^i \mid i = 1, \cdots, n\}$ 张成 n 维闭凸集. □

由命题 20.4.2 可知, 对每个本质博弈都存在无穷多个不同的分配. 其实这时 $E(v)$ 是一个不可数集. 于是, 选择最合理的分配就成为一个合乎逻辑的命题.

下面的定义表示一个给定联盟 R 的局中人对分配优劣的判断.

定义 20.4.2 设 x, y 为 (N,v) 的两个分配. $\varnothing \neq R \in 2^N$. 称 x 关于 R 优超 (dominate) y, 记作 $x \succ_R y$, 如果

(i)

$$x_i > y_i, \quad \forall i \in R; \tag{20.4.3}$$

(ii)

$$v(R) \geqslant \sum_{i \in R} x_i. \tag{20.4.4}$$

条件 (20.4.3) 表明 R 中人都认为 x 比 y 好, 因此会选择 x. 条件 (20.4.4) 表明 x 是可实现的分配, 不是空头支票.

命题 20.4.3 对于单点集 $R = \{i\}$, 或大联盟 $R = N$, 不存在关于 R 的优超集.

证明 (i) 设 $R = \{i\}$: 若 $x \succ_R y$, 则 $x_i > y_i \geqslant v(\{i\})$; 并且 $x_i \leqslant v(\{i\})$. 于是有

$$y_i < v(\{i\}),$$

这与 y 为分配矛盾.

(ii) 设 $R = N$: 若 $x \succ_R y$, 则 $x_i > y_i$, $\forall i$. 但 $\sum_{i=1}^n y_i = v(N)$, 则 $\sum_{i=1}^n x_i > v(N)$, 与分配的定义矛盾. □

如果存在 $R \neq \varnothing$, 使 x 关于 R 优超 y, 即 $x \succ_R y$, 则简称 x 优超 y, 记作 $x \succ y$. 容易证明, 关于同一子集 R 的优超具有传递性: 即

$$x \succ_R y, \quad y \succ_R z \implies x \succ_R z.$$

但优超没有传递性: 即

$$x \succ y, \quad y \succ z \not\Longrightarrow x \succ z.$$

定义 20.4.3 设 (N,u) 与 (N,v) 为两个合作博弈. u 与 v 称为同构博弈 (isomorphic game), 如果存在一个双向一对一映射 $f: E(u) \to E(v)$, 称为同构映射, 使得

$$x \succ_R y \Longleftrightarrow f(x) \succ_R f(y), \quad x, y \in E(u). \tag{20.4.5}$$

下面的定理说明: “同构” 与 “策略等价” 是一致的.

定理 20.4.1 ([12]) 在 N 上的两特征函数 u 与 v 为同构博弈, 当且仅当 u 与 v 策略等价 (即 $u \sim v$).

20.5 核　心

合作博弈的根本问题是要找到一个最合理的分配. 对于这个问题许多人给出了不同的答案. 但是, 不像纳什均衡被广泛接受为非合作博弈的解那样, 至今尚没有一个在各种情况下都普适的 “最佳分配”.

本节所讨论的核心是其中的一种分配, 它的合理性是基于分配的优超概念. 在介绍这个概念前, 先做一点记号的准备.

设 $x \in E(v)$, 定义

$$x(R) := \sum_{i \in R} x_i, \quad R \in 2^N. \tag{20.5.1}$$

实际上, (20.5.1) 使 x 变成了一个特征函数, 只不过它所对应的是一个平凡博弈. 将 2^n 个连续整数 $2^n-1,\ 2^n-2,\ \cdots, 1, 0$ 写成 n 位的 2 进制数, 得

$$b_1 = 1, \cdots, 1, 1, 1;\ b_2 = 1, \cdots, 1, 1, 0;\ b_3 = 1, \cdots, 1, 0, 1;\ \cdots;\ b_{2^n} = 0, \cdots, 0, 0, 0.$$

它们分别为

$$R_1 = N;\ R_2 = N \backslash \{n\};\ R_3 = N \backslash \{n-1\};\ \cdots;\ R_{2^n} = \varnothing,$$

的结构向量. 构造矩阵

$$M_n := \left[b_1^{\mathrm{T}},\ b_2^{\mathrm{T}},\ b_3^{\mathrm{T}},\ \cdots,\ b_{2^n}^{\mathrm{T}}\right] \in \mathcal{B}_{n \times 2^n}. \tag{20.5.2}$$

那么, 所生成的特征函数的结构向量, 记作 V_x, 可表示为

$$V_x = x M_n. \tag{20.5.3}$$

定义 20.5.1 合作博弈 (N, v) 的不被任何分配所优超的分配的全体称为其核心 (core), 记作 $C(v)$.

定理 20.5.1 设 (N, v) 为合作博弈, $x \in \mathbb{R}^n$. 如果

(i)

$$x(R) \geqslant v(R), \quad \forall R \in 2^N; \tag{20.5.4}$$

(ii)

$$x(N)=v(N), \tag{20.5.5}$$

则 $x \in C(v)$.

设 v 满足超可加性, 则条件 (i)–(ii) 也是 $x \in C(v)$ 的必要条件.

证明 设 $x \in \mathbb{R}^N$ 满足 (20.5.4) 及 (20.5.5), 则 $x \in E(v)$. 设 $x \notin C(v)$, 则存在 $\varnothing \neq R \in 2^N$ 及 $y \in E(v)$, 使得

$$\begin{aligned}
&y_i > x_i, \quad i \in R;\\
&v(R) \geqslant y(R).
\end{aligned}$$

于是有

$$v(R) \geqslant y(R) > x(R),$$

这与 (20.5.4) 矛盾.

下面设 v 满足超可加性, 证明必要性: 设 $x \in C(v)$, 则 $x \in E(v)$. 于是有 (20.5.5). 假定 (20.5.4) 不成立, 则存在 $R \in 2^N$, 使 $x(R) < v(R)$. 显然, $R \neq \varnothing$ 且 $R \neq N$. 定义

$$\begin{aligned}
\alpha &= \frac{v(R)-x(R)}{|R|} > 0;\\
\beta &= \frac{1}{n-|R|}\left(v(N)-v(R)-\sum_{i\in R^c} v(\{i\})\right) \geqslant 0.
\end{aligned}$$

然后定义 $y \in \mathbb{R}^n$ 如下:

$$y_i = \begin{cases} x_i+\alpha, & i \in R,\\ v(\{i\})+\beta, & i \notin R. \end{cases}$$

容易验证: $y \in E(v)$, 且 $y \succ_R x$, 它与 $x \in C(v)$ 矛盾. □

注意, 利用式 (20.5.3), 式 (20.5.4) 可表为矩阵形式为

$$M_n^{\mathrm{T}} x^{\mathrm{T}} \geqslant V_v^{\mathrm{T}}. \tag{20.5.6}$$

但第一个方程应用等式 (20.5.5) 代替, 最后一个方程是恒等式. 将 M_n^{T} 的第一行与最后一行去掉, 记余下的矩阵为 N_n, 同样, 将 V_v^{T} 的第一行与最后一行去掉, 记余下的向量为 W_v. 那么, 寻求核心就是要求解

$$\begin{cases} \displaystyle\sum_{i=1}^{n} x_i = v(N),\\ N_n x^{\mathrm{T}} \geqslant W_v. \end{cases} \tag{20.5.7}$$

例 20.5.1 回忆例 20.1.2 中的买卖马问题. 有

$$M_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}; \quad V_v = [1100, 1000, 1100, 0, 0, 0, 0, 0].$$

于是, 方程 (20.5.7) 变为

$$\begin{cases} x_1 + x_2 + x_3 = 1100, \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \geqslant \begin{bmatrix} 1000 \\ 1100 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \end{cases} \tag{20.5.8}$$

解得

$$\begin{cases} x_1 \in [1000,\ 1100], \\ x_2 = 0, \\ x_3 = 1100 - x_1. \end{cases}$$

于是

$$C(v) = \{(t, 0, 1100 - t) \mid 1000 \leqslant t \leqslant 1100\}.$$

核心是一种非常合理的分配方案, 因为没有一个联盟能找到比它更好的 (优超的) 分配方案. 但它的致命弱点是, 这种分配常常不存在. 例如

定理 20.5.2 设 (N, v) 为一常和的本质博弈, 则

$$C(v) = \varnothing.$$

证明 设 $C(v) \neq \varnothing$, 则存在 $x \in C(v)$. 于是有

$$v(\{i\}^c) \leqslant x(\{i\}^c), \quad \forall i \in N.$$

由于 (N, v) 是常和的, 由互补性

$$v(N) = v(\{i\}) + v(\{i\}^c), \quad \forall i \in N.$$

于是有

$$\begin{aligned} v(N) &= x(N) \\ &= \sum_{i=1}^{n} (x(N) - x(\{i\}^c)) \\ &\leqslant \sum_{i=1}^{n} (x(N) - v(\{i\}^c)) \\ &= \sum_{i=1}^{n} v(\{i\}). \end{aligned}$$

这与本质博弈相矛盾. 故 $C(v) = \varnothing$. □

20.6 核心的存在性

本节讨论几类特殊的合作博弈, 探讨何时其核心存在.

20.6.1 简单博弈

定义 20.6.1 合作博弈 (N, v) 称为一个简单博弈, 如果 (i) $v(\{i\}) = 0, \forall i \in N$; (ii) $v(N) = 1$; (iii) $v(R) = 0$ 或 $1, \forall R \in 2^N$.

简单博弈模型来自社会或者说政治行为. 取值为 1 的联盟称为取胜联盟 (winning coalition), 取值为 0 的联盟称为失败联盟 (losing coalition). $v(R)$ 表示联盟 R 的胜负.

简单博弈可细分成几类:

(i) 加权多数 (weighted majority) 博弈: (N, v) 中每位玩家 i 有 p_i 张选票. 总票数达到 q 以上则胜, 不到则负. 于是有

$$v(R) = \begin{cases} 0, & \sum_{i \in R} p_i < q, \\ 1, & \sum_{i \in R} p_i \geqslant q. \end{cases} \tag{20.6.1}$$

为体现合理性, 通常要求 $\frac{1}{2} \sum_{i \in N} p_i < q \leqslant \sum_{i \in N} p_i$.

(ii) 简单多数 (simple majority) 博弈: 权重 $p_i = 1, \forall i \in N$ 的加权多数对策称为简单多数博弈.

(iii) 一票否决 (one vote veto) 博弈: 在简单多数博弈中, 如果 $q = n$, 则成一票否决博弈.

由定义容易证明以下命题:

命题 20.6.1 简单博弈的特征函数具有单调性. 即

$$R \subset T \Rightarrow v(R) \leqslant v(T).$$

定义 20.6.2 在一个简单博弈 (N,v) 中, 如果存在 i, 满足 $v(\{i\}^c)=0$. 则 i 称为否决人 (veto player).

定理 20.6.1 设 (N,v) 为简单博弈. 则 $C(v)\neq\varnothing$ 当且仅当存在否决人.

证明 (充分性) 设 i_0 为否决人. 定义

$$x_i = \begin{cases} 1, & i = i_0, \\ 0, & i \in \{i_0\}^c. \end{cases}$$

则 $x=(x_1,\cdots,x_n)\in C(v)$. 反设 $x\notin C(v)$. 由定理 20.5.1, 则存在 R 使 $v(R) > x(R)$. 由 $x(R)\geqslant 0$, 则 $v(R)>0$. 由于 (N,v) 是简单博弈, 因此, $v(R)=1$. 从而 $x(R)<1$, 故 $i_0\notin R$. 于是 $R\subset\{i_0\}^c$, 由单调性 (见命题 20.6.1) 有

$$v(\{i_0\}^c)\geqslant v(R)=1.$$

但 i_0 是否决人, 故 $v(\{i_0\}^c)=0$, 矛盾. 故 $x\in C(v)$.

(必要性) 设 $C(v)\neq\varnothing$, 但不存在否决人. 即

$$v(\{i\}^c)=1,\quad \forall i\in N.$$

设 $x\in C(v)$, 则

$$\begin{aligned} &x(N)=v(N)=1, \\ &x(\{i\}^c)\geqslant v(\{i\}^c)=1,\quad \forall i\in N. \end{aligned}$$

于是有

$$\begin{aligned} x_i &= x(N)-x(\{i\}^c) \\ &\leqslant v(N)-v(\{i\}^c) \\ &=0,\quad \forall i\in N. \end{aligned}$$

即 $x_i\leqslant 0, \forall i\in N$. 于是, $x_{i_0}\leqslant 0$, 矛盾. □

根据上述定理, 一票否决博弈的核心非空.

20.6.2　凸合作博弈

定义 20.6.3　一个合作博弈 (N,v) 称为凸合作博弈, 如果它满足

$$v(R)+v(T)\leqslant v(R\cup T)+v(R\cap T),\quad \forall R,T\in 2^N. \tag{20.6.2}$$

定理 20.6.2　凸合作博弈的核心非空.

证明　记 $N=\{1,2,\cdots,n\}$. 令

$$\begin{aligned}&x_1=v(\{1\}),\\&x_i=v(\{1,2,\cdots,i\})-v(\{1,2,\cdots,i-1\}),\quad i=2,\cdots,n.\end{aligned}$$

下面证明 $x\in C(v)$. 显然 $x(N)=v(N)$. 设 $R\in 2^N$, 记

$$R^c=\{j_1,j_2,\cdots,j_t\},\quad j_1<j_2<\cdots<j_t.$$

令 $T=\{1,2,\cdots,j_1\}$, 则有

$$R\cup T=R\cup\{j_1\},\quad R\cap T=T\backslash\{j_1\}.$$

利用凸性, 可得

$$v(R)+v(T)\leqslant v(R\cup\{j_1\})+v(T\backslash\{j_1\}). \tag{20.6.3}$$

由定义及 (20.6.3) 知

$$x_{j_1}=v(T)-v(T\backslash\{j_1\})\leqslant v(R\cup\{j_1\})-v(R).$$

也就是

$$x(R\cup\{j_1\})-x(R)\leqslant v(R\cup\{j_1\})-v(R).$$

所以

$$x(R)-v(R)\geqslant x(R\cup\{j_1\})-v(R\cup\{j_1\}). \tag{20.6.4}$$

用 $R\cup\{j_1\}$ 代替 R, j_2 代替 j_1, (20.6.4) 变为

$$x(R\cup\{j_1\})-v(R\cup\{j_1\})\geqslant x(R\cup\{j_1,j_2\})-v(R\cup\{j_1,j_2\}).$$

重复 t 次即得

$$x(R)-v(R)\geqslant x(N)-v(N)=0.$$

由定理 20.5.1 即得结论. □

20.6.3 严对称博弈

定义 20.6.4 一个合作博弈 (N,v) 称为严对称博弈, 如果它满足: 若 $|R|=|T|$, 则 $v(R)=v(T)$.

定理 20.6.3 设合作博弈 (N,v) 是严对称的. 那么, $C(v)\neq\varnothing$ 当且仅当

$$\frac{v(R)}{|R|}\leqslant\frac{v(N)}{|N|},\quad\forall\varnothing\neq R\in 2^N. \tag{20.6.5}$$

证明 (充分性) 设 (20.6.5) 成立. 定义

$$x=\left(\frac{v(N)}{|N|},\frac{v(N)}{|N|},\cdots,\frac{v(N)}{|N|}\right),$$

则 $x(N)=v(N)$.

如果 $R=\varnothing$ 显见, $x(R)=v(R)=0$. 设 $R\neq\varnothing$. 由 (20.6.5) 可得

$$v(R)\leqslant\frac{|R|}{|N|}v(N)=x(R).$$

故 $x\in C(v)$, 因此, $C(v)\neq\varnothing$.

(必要性) 反设存在 $\varnothing\neq R_0\subsetneq N$, 使得

$$\frac{v(R_0)}{|R_0|}>\frac{v(N)}{|N|}.$$

记 $|R_0|=r$. 任选 $x\in E(v)$. 记 x 的 r 个最小分量为 $\{x_{i_1},x_{i_2},\cdots,x_{i_r}\}$. 定义 $T=\{i_1,i_2,\cdots,i_r\}$. 那么

$$\frac{1}{r}x(T)\leqslant\frac{1}{|N|}x(N)=\frac{1}{|N|}v(N).$$

即

$$x(T)\leqslant\frac{r}{|N|}v(N).$$

因为 (N,v) 是严对称的, 所以

$$v(T)=v(R_0)>\frac{r}{|N|}v(N)\geqslant x(T).$$

于是, $x\notin C(v)$. 但 $x\in E(v)$ 是任选的, 故 $C(v)=\varnothing$. □

例 20.6.1 设有 n 支手套, 不分左右手. 将其配套出售. 将其看作 n 人合作博弈, 求其核心.

将所配手套副数作为收益, 则有

$$v(R)=\begin{cases}\dfrac{|R|}{2}, & |R| \text{ 为偶数},\\[2ex] \dfrac{|R|-1}{2}, & |R| \text{ 为奇数}.\end{cases}$$

显见, 这是一个严对称博弈.

当 n 为奇数, 取 $R_0 \subset N$, $|R_0| = n-1$. 则

$$\frac{v(R_0)}{|R_0|} > \frac{v(N)}{|N|}.$$

由定理 20.6.3 可知: $C(v) = \varnothing$.

当 n 为偶数, 不难检验 (20.6.5) 对所有 $R \neq \varnothing$ 均成立, 于是 $C(v) \neq \varnothing$. 设 $x \in C(v)$, 则应满足

$$\sum_{i=1}^{n} x_i = \frac{n}{2},$$

但

$$x_i + x_j \geqslant 1, \quad i \neq j,$$

则得

$$x_i + x_j = 1, \quad i \neq j.$$

显然, 应有

$$x_i = \frac{1}{2}, \quad \forall i \in N.$$

因此, 唯一可能解为

$$C(v) = \left\{\left(\frac{1}{2}, \frac{1}{2}, \cdots, \frac{1}{2}\right)\right\}.$$

20.7 Shapley 值

虽然, 现在对于合作博弈的解 (即分配) 已有许多方案, 但 Shapley 值与核心是两个用得最多的分配. Shapley 值的优点之一是它存在且唯一. 而其合理性表现在它满足三个公理.

考虑合作博弈 (N, v). 下面这个分配来自一个很自然的想法:

$$\begin{aligned}
x_1 &= v(\{1\}),\\
x_2 &= v(\{1,2\}) - v(\{1\}),\\
x_3 &= v(\{1,2,3\}) - v(\{1,2\}),\\
&\ \vdots\\
x_n &= v(\{1,2,\cdots,n\}) - v(\{1,2,\cdots,n-1\}).
\end{aligned}$$

它的一个问题是, 这种分配依赖于 N 中玩家的排序. 那么, 换一下排序, 令 $\sigma \in \mathbf{S}_n$ 为一置换. 于是, 在 $\sigma(i)$ 的顺序下, 得到另一个分配:

$$\begin{aligned}
x_1 &= v(\{\sigma^{-1}(1)\}),\\
x_2 &= v(\{\sigma^{-1}(1), \sigma^{-1}(2)\}) - v(\{\sigma^{-1}(1)\}),\\
x_3 &= v(\{\sigma^{-1}(1), \sigma^{-1}(2), \sigma^{-1}(3)\}) - v(\{\sigma^{-1}(1), \sigma^{-1}(2)\}),\\
&\ \ \vdots\\
x_n &= v(\{\sigma^{-1}(1), \sigma^{-1}(2), \cdots, \sigma^{-1}(n)\})\\
&\quad - v(\{\sigma^{-1}(1), \sigma^{-1}(2), \cdots, \sigma^{-1}(n-1)\}),
\end{aligned}$$

这里, $\sigma^{-1}(i)$ 指现在排在第 i 位的玩家.

对每一个 $\sigma \in \mathbf{S}_n$, 定义在这个排列中排在玩家 i 前面的玩家记为

$$S_\sigma^i = \{j | \sigma(j) < \sigma(i)\}.$$

对 $\mathbf{S}_n$ 上所有置换取平均, 则得

$$\varphi_i(v) := \frac{1}{n!} \sum_{\sigma \in \mathbf{S}_n} \left[v(S_\sigma^i \cup \{i\}) - v(S_\sigma^i) \right], \quad i = 1, \cdots, n. \tag{20.7.1}$$

显然有

$$\begin{cases} \displaystyle\sum_{i=1}^n \varphi_i(v) = v(N), \\ \varphi_i(v) \geqslant v(\{i\}), \quad i = 1, \cdots, n. \end{cases} \tag{20.7.2}$$

于是, $\varphi := (\varphi_1(v), \varphi_2(v), \cdots, \varphi_n(v)) \in E(v)$ 是一个分配.

下面, 将 (20.7.1) 右边各项按 $S \in N \backslash i$ 分类. 定义

$$\Theta^S := \left\{ \sigma \in \mathbf{S}_n \middle| S_\sigma^i = S \right\}.$$

注意, 元素在 S_σ^i 中的序号不会影响 φ_i 的定义. 现在有

$$|\Theta^S| = |S|!(n-1-|S|)!.$$

因此, 可得到

$$\begin{aligned}
\varphi_i(v) &= \frac{1}{n!} \sum_{S \subset N \backslash \{i\}} \sum_{\sigma \in \Theta^S} \left[v\left(S_\sigma^i \cup \{i\}\right) - v\left(S_\sigma^i\right) \right]\\
&= \sum_{S \subset N \backslash \{i\}} \frac{|S|!(n-1-|S|)!}{n!} \left[v\left(S \cup \{i\}\right) - v\left(S\right) \right],\\
&\qquad\qquad i = 1, 2, \cdots, n.
\end{aligned} \tag{20.7.3}$$

定义 20.7.1　分配

$$\varphi = (\varphi_1, \varphi_2, \cdots, \varphi_n) \in E(v)$$

称为 Shapley 值.

定义 20.7.2　考虑合作博弈 (N, v), 令 $T \in 2^N$. T 称为 v 的一个支柱 (carrier), 如果

$$v(R) = v(R \cap T), \quad \forall R \in 2^N.$$

命题 20.7.1　1) 设 T 为支柱, 且 $T \subset W \subset N$. 则 W 也是支柱.

2) 设 T 为支柱, 且 $i \notin T$. 那么

$$v(R \cup \{i\}) = v(R), \quad \forall R \in 2^N.$$

这时, i 称为一个哑玩家 (dummy).

证明　1)

$$\begin{aligned} v(R \cap W) &= v((R \cap W) \cap T) \\ &= v(R \cap T) \\ &= v(R). \end{aligned}$$

2)

$$\begin{aligned} v(R \cup \{i\}) &= v((R \cup \{i\}) \cap T) \\ &= v(R \cap T) \\ &= v(R). \end{aligned}$$

□

定义 20.7.3　考查合作博弈 (N, v). 定义映射 $\psi : v \to E(v)$ 如下:

$$\psi(v) = (\psi_1(v), \psi_2(v), \cdots, \psi_n(v)).$$

三个基本公理定义为

1) 有效性公理 (efficiency axiom): 对 v 的支柱 T,

$$\sum_{i \in T} \psi_i(v) = v(T). \tag{20.7.4}$$

2) 对称公理 (symmetry axiom): 对任一排列 $\sigma \in \mathbf{S}_n$, 使得

$$v(\sigma(R)) = v(R), \quad \forall R \in 2^N, \tag{20.7.5}$$

则有

$$\psi_{\sigma(i)}(v) = \psi_i(v), \quad \forall i \in N. \tag{20.7.6}$$

3) 可加性公理 (additivity axiom): 设 v, w 为 N 上的两个特征函数. 那么

$$\psi_i(v+w) = \psi_i(v) + \psi_i(w), \quad \forall i \in N. \tag{20.7.7}$$

定理 20.7.1 由 (20.7.3) 所定义的 Shapley 值满足三个基本公理.

证明 1) (有效性公理) 设 T 为 v 的一个支柱, 那么

$$v(R \cup \{i\}) = v(R), \quad \forall i \in T^c, \forall R \in 2^N.$$

因此, 对所有的 $i \in T^c$ 有

$$\varphi_i(v) = \sum_{R \subset \{i\}^c} \frac{|R|!(n-1-|R|)!}{n!} [v(R \cup \{i\}) - v(R)] = 0.$$

根据 (20.7.2) 有

$$\begin{aligned} v(T) &= v(N \cap T) \\ &= v(N) \\ &= \sum_{i \in N} \varphi_i(v) \\ &= \sum_{i \in T} \varphi_i(v). \end{aligned}$$

2) (对称性公理) 设 $\sigma \in \mathbf{S}_n$ 满足 (20.7.5). 那么, $|\sigma(R)| = |R|$, $\forall R \in 2^N$, 而且, 对任何 $i \in N$, 有

$$\sigma(R) \subset N \backslash \{\sigma(i)\} \Leftrightarrow R \subset N \backslash \{i\}.$$

利用 (20.7.5), 有

$$\begin{aligned} \varphi_{\sigma(i)}(v) &= \sum_{\sigma(R) \subset N \backslash \{\sigma(i)\}} \frac{|\sigma(R)|!(n-1-|\sigma(R)|)!}{n!} [v(\sigma(R) \cup \{\sigma(i)\}) - v(\sigma(R))] \\ &= \sum_{R \subset N \backslash \{i\}} \frac{R|!(n-1-|R|)!}{n!} [v(R \cup \{i\}) - v(R)] \\ &= \varphi_i(v), \quad \forall i \in N. \end{aligned}$$

3) (可加性公理) 因为 $\varphi_i(v)$ 是 v 的一个线性函数, $\forall i$, 显然 $\varphi(v)$ 满足可加性公理. □

引理 20.7.1 考虑一个合作博弈 (N,v). u_T, $\varnothing \neq T \in 2^N$, 为一无异议博弈, $c \geqslant 0$. 设 $\psi : v \to E(v)$ 满足三个基本公理. 那么

$$\psi_i(c\, u_T) = \begin{cases} 0, & i \notin T, \\ \dfrac{c}{|T|}, & i \in T. \end{cases} \tag{20.7.8}$$

证明 容易检验 $c\, u_T$ 是一个特征函数且 T 是 $c\, u_T$ 的一个支柱. 设 $i \in N \backslash T$. 由于 T 和 $T \cup \{i\}$ 均为 $c\, u_T$ 的支柱, 根据有效性公理, 有

$$\begin{aligned} \sum_{j\in T} \psi_j(c\, u_T) &= c\, u_T(T) \\ &= c\, u_T(T \cup \{i\}) \\ &= \sum_{j\in T\cup\{i\}} \psi_j(c\, u_T) \\ &= \sum_{j\in T} \psi_j(c\, u_T) + \psi_i(c\, u_T). \end{aligned}$$

于是

$$\psi_i(c\, u_T) = 0, \quad \forall i \notin T.$$

其次, 设 $i, j \in T$, $i \neq j$. 令 $\sigma = (i, j)$. 先检验 (20.7.5). 即

$$c\, u_T(\sigma(R)) = c\, u_T(R), \quad \forall R \in 2^N. \tag{20.7.9}$$

• 情况 1: $T \subset R$. 那么 $T = \sigma(T) \subset \sigma(R)$. 因此

$$c\, u_T(\sigma(R)) = c = c\, u_T(R).$$

• 情况 2: $T \not\subset R$. (i) $i \notin R$, 那么 $j \notin \sigma(R) \Rightarrow T \not\subset \sigma(R)$; (ii) $j \notin R$, 那么 $i \notin \sigma(R)$ $\Rightarrow T \not\subset \sigma(R)$; (iii) $i, j \in R$, 则存在 $k \in T \backslash \{i, j\}$, $k = \sigma(k) \notin \sigma(R) \Rightarrow T \not\subset \sigma(R)$. 因此,

$$c\, u_T(\sigma(R)) = c = c\, u_T(R).$$

由对称性公理, 有

$$\psi_j(c\, u_T) = \psi_{\sigma(i)}(c\, u_T) = \psi_i(c\, u_T).$$

利用有效性公理, 有

$$|T|\psi_i(c\, u_T) = \sum_{i\in T} \psi_i(c\, u_T) = c\, u_T(T) = c, \quad \forall i \in T.$$

$$\psi_i(c\, u_T) = \frac{c}{|T|}, \quad \forall i \in T.$$

□

定理 20.7.2 由 (20.7.3) 所定义的 Shapley 值是唯一满足三个基本公理的分配.

证明 定理 20.7.1 表明 Shapley 满足三个基本公理. 下面证明唯一性. 利用无异议基底, 有

$$\begin{aligned} v &= \sum_{\varnothing \neq T \in 2^N} c_T u_T \\ &= \sum_{\substack{\varnothing \neq T \in 2^N \\ c_T \geqslant 0}} c_T u_T - \sum_{\substack{\varnothing \neq T \in 2^N \\ c_T < 0}} (-c_T) u_T. \end{aligned}$$

由可加性公理, 对任何满足三个基本公理的分配 ψ 均有

$$\psi_i(v) = \sum_{\substack{\varnothing \neq T \in 2^N \\ c_T \geqslant 0}} \psi_i(c_T u_T) - \sum_{\substack{\varnothing \neq T \in 2^N \\ c_T < 0}} \psi_i(-c_T u_T), \quad \forall i \in N.$$

利用引理 20.7.1, 有

$$\begin{aligned} \psi_i(v) &= \sum_{\substack{i \in T \in 2^N \\ c_T \geqslant 0}} \frac{c_T}{|T|} - \sum_{\substack{i \in T \in 2^N \\ c_T < 0}} \frac{-c_T}{|T|} \\ &= \sum_{i \in T \in 2^N} \frac{c_T}{|T|}. \end{aligned}$$

因此, $\psi_i(v)$ 由 v, N 和 i 唯一决定, 这表明 Shapley 值是唯一满足三个基本公理的分配. □

下面给一个计算 Shapley 值的简洁公式. 注意到

$$\begin{aligned} v(R \cup \{i\}) - v(R) &= C_v \left[x_1^R \cdots x_{i-1}^R \begin{pmatrix} 1 \\ 0 \end{pmatrix} x_{i+1}^R \cdots x_n^R - x_1^R \cdots x_{i-1}^R \begin{pmatrix} 0 \\ 1 \end{pmatrix} x_{i+1}^R \cdots x_n^R \right] \\ &= C_v \left[W_{[2,2^{i-1}]} \begin{pmatrix} 1 \\ 0 \end{pmatrix} x_1^R \cdots x_{i-1}^R x_{i+1}^R \cdots x_n^R \right. \\ &\quad \left. - W_{[2,2^{i-1}]} \begin{pmatrix} 0 \\ 1 \end{pmatrix} x_1^R \cdots x_{i-1}^R x_{i+1}^R \cdots x_n^R \right] \\ &= C_v \left[W_{[2,2^{i-1}]} \begin{pmatrix} 1 \\ -1 \end{pmatrix} x_1^R \cdots x_{i-1}^R x_{i+1}^R \cdots x_n^R \right], \end{aligned} \tag{20.7.10}$$

这里,

$$x_j^R = \begin{cases} \delta_2^1, & j \in R, \\ \delta_2^2, & j \notin R. \end{cases}$$

下面定义

$$|\delta_{2^k}^i| := |R|,$$

这里 $x^R = \ltimes_{j=1}^n x_j^R = \delta_{2^k}^i$. 那么, 容易证明以下引理.

引理 20.7.2　构造一组列向量

$$\begin{cases} \ell_1 & = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \in \mathbb{R}^2; \\ \ell_{k+1} & = \begin{bmatrix} \ell_k + \mathbf{1}_{2^k} \\ \ell_k \end{bmatrix} \in \mathbb{R}^{2^{k+1}}, \\ & \qquad k = 2, 3, \cdots, \end{cases}$$

这里 $\mathbf{1}_t = \underbrace{[1, \cdots, 1]}_{t}^{\mathrm{T}}$. 于是

$$\left|\delta_{2^k}^i\right| = \ell_k^i, \quad i = 1, 2, \cdots, 2^k. \tag{20.7.11}$$

(这里 ℓ_k^i 是 ℓ_k 的第 i 个分量.)

利用 ℓ_k, 构造一个列向量 $\zeta_k \in \mathbb{R}^{2^k}$ 如下:

$$\zeta_k^i = \left(\ell_k^i\right)! \left(k - \ell_k^i\right)!, \quad i = 1, \cdots, 2^k. \tag{20.7.12}$$

利用 (20.7.10) 和 (20.7.12), (20.7.3) 可以写成

$$\varphi_i(v) = \frac{1}{n} V_v \sum_{j=1}^{2^{n-1}} \zeta_{n-1}^j W_{[2,2^{i-1}]} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \delta_{2^{n-1}}^j, \quad i = 1, \cdots, n. \tag{20.7.13}$$

注意到

$$W_{[2,2^{i-1}]} = \delta_{2^i}\left[1, 3, \cdots, 2^{i-1} - 1, 2, 4, \cdots, 2^{i-1}\right],$$

那么

$$W_{[2,2^{i-1}]} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \underbrace{\begin{bmatrix} \binom{1}{-1} & 0 & \cdots & 0 \\ 0 & \binom{1}{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \binom{1}{-1} \end{bmatrix}}_{2^{i-1}}.$$

下面构造一个矩阵 $\Gamma_i \in \mathcal{M}_{2^n\times 2^{n-1}}$ 如下:

$$
\begin{aligned}
\Gamma_i &= \left[W_{[2,2^{i-1}]}\binom{1}{-1}\right]\otimes I_{2^{n-i}} \\
&= \underbrace{\begin{bmatrix}
\begin{pmatrix} I_{2^{n-i}} \\ -I_{2^{n-i}} \end{pmatrix} & 0 & \cdots & 0 \\
0 & \begin{pmatrix} I_{2^{n-i}} \\ -I_{2^{n-i}} \end{pmatrix} & \cdots & 0 \\
\vdots & \vdots & & \vdots \\
0 & 0 & \cdots & \begin{pmatrix} I_{2^{n-i}} \\ -I_{2^{n-i}} \end{pmatrix}
\end{bmatrix}}_{2^{i-1}}.
\end{aligned}
$$

显然

$$
W_{[2,2^{i-1}]}\begin{pmatrix}1\\-1\end{pmatrix}\delta_{2^{n-1}}^j = \mathrm{Col}_j(\Gamma_i).
$$

定义一个新向量

$$
\eta := \zeta_{n-1}. \tag{20.7.14}
$$

然后将 η 等分成 k 块

$$
\eta = \begin{bmatrix}\eta_k^1\\ \eta_k^2 \\ \vdots \\ \eta_k^k\end{bmatrix}, \quad k = 1, 2, 2^2, \cdots, 2^{n-1}.
$$

根据 Γ_i 的构造, 不难证明

$$
\sum_{j=1}^{2^{n-1}} \zeta_{n-1}^j W_{[2,2^{i-1}]}\begin{pmatrix}1\\-1\end{pmatrix}\delta_{2^{n-1}}^j = \begin{bmatrix}
\eta_{2^{i-1}}^1 \\ -\eta_{2^{i-1}}^1 \\ \eta_{2^{i-1}}^2 \\ -\eta_{2^{i-1}}^2 \\ \vdots \\ \eta_{2^{i-1}}^{2^{i-1}} \\ -\eta_{2^{i-1}}^{2^{i-1}}
\end{bmatrix}.
$$

综合以上的构造和讨论可知

定理 20.7.3　合作博弈 (N,v) (这里 $|N|=n$) 的 Shapley 值可计算如下:

$$V_v \Xi_n = \varphi(v), \tag{20.7.15}$$

这里 $\Xi_n \in \mathcal{M}_{2^n\times n}$ 为

$$\Xi_n = \frac{1}{n!}\begin{bmatrix} \begin{pmatrix} \eta_1 \\ -\eta_1 \end{pmatrix} & \begin{pmatrix} \eta_2^1 \\ -\eta_2^1 \\ \eta_2^2 \\ -\eta_2^2 \end{pmatrix} & \begin{pmatrix} \eta_4^1 \\ -\eta_4^1 \\ \eta_4^2 \\ -\eta_4^2 \\ \eta_4^3 \\ -\eta_4^3 \\ \eta_4^4 \\ -\eta_4^4 \end{pmatrix} & \cdots & \begin{pmatrix} \eta_{2^{n-1}}^1 \\ -\eta_{2^{n-1}}^1 \\ \eta_{2^{n-1}}^2 \\ -\eta_{2^{n-1}}^2 \\ \vdots \\ \eta_{2^{n-1}}^{2^{n-1}} \\ -\eta_{2^{n-1}}^{2^{n-1}} \end{pmatrix} \end{bmatrix}. \tag{20.7.16}$$

例 20.7.1　1) $n=2$:

$$\ell_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}^{\mathrm{T}};$$

$$\eta_1 = \begin{bmatrix} 1!(2-1-1)! & 0!(2-1-0)! \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} 1 & 1 \end{bmatrix}^{\mathrm{T}};$$

$$\Xi_2 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 1 \\ -1 & -1 \end{bmatrix}.$$

2) $n=3$:

$$\ell_2 = \begin{bmatrix} 2 & 1 & 1 & 0 \end{bmatrix}^{\mathrm{T}};$$

$$\eta_2 = \begin{bmatrix} 2 & 1 & 1 & 2 \end{bmatrix}^{\mathrm{T}};$$

$$\Xi_3 = \frac{1}{6}\begin{bmatrix} 2 & 2 & 2 \\ 1 & 1 & -2 \\ 1 & -2 & 1 \\ 2 & -1 & -1 \\ -2 & 1 & 1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \\ -2 & -2 & -2 \end{bmatrix}.$$

3) $n = 4$:

$$\ell_3 = \begin{bmatrix} 3 & 2 & 2 & 1 & 2 & 1 & 1 & 0 \end{bmatrix}^{\mathrm{T}};$$

$$\eta_3 = \begin{bmatrix} 6 & 2 & 2 & 6 & 2 & 6 & 6 & 6 \end{bmatrix}^{\mathrm{T}};$$

$$\varXi_4 = \frac{1}{24}\begin{bmatrix} 6 & 6 & 6 & 6 \\ 2 & 2 & 2 & -6 \\ 2 & 2 & -6 & 2 \\ 6 & 6 & -2 & -2 \\ 2 & -6 & 2 & 2 \\ 6 & -2 & 6 & -2 \\ 6 & -2 & -2 & 6 \\ 6 & -6 & -6 & -6 \\ -6 & 2 & 2 & 2 \\ -2 & 6 & 6 & -2 \\ -2 & 6 & -2 & 6 \\ -6 & 6 & -6 & -6 \\ -2 & -2 & 6 & 6 \\ -6 & -6 & 6 & -6 \\ -6 & -6 & -6 & 6 \\ -6 & -6 & -6 & -6 \end{bmatrix}.$$

例 20.7.2 回忆例 20.1.2 和例 20.5.1 中的买卖马问题.

$$V_v = \begin{bmatrix} 1100 & 1000 & 1100 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

利用公式 (20.7.15), 则 Shapley 值为

$$\varphi(v) = V_v \varXi_3 = \begin{bmatrix} 716.7 & 166.7 & 216.7 \end{bmatrix}.$$

20.8 Shapley 值与核心的关系

比较例 20.7.2 与例 20.5.1, 我们发现, 对于买卖马问题 Shapley 值与核心相去甚远. 这说明 Shapley 值虽然存在唯一, 但有时与合理的解有距离. 因此, 一个合理的问题是: 什么时候 Shapley 值也是核心? 一个充分条件是:

定理 20.8.1 设 (N, v) 为一凸合作博弈, 则 Shapley 值

$$\varphi(v) \in C(v).$$

证明　对任一 $\sigma \in \mathbf{S}_n$, 令

$$x_\sigma^i := v\left(S_\sigma^i \cup \{i\}\right) - v\left(S_\sigma^i\right), \quad i = 1, \cdots, n.$$

于是可得

$$x_\sigma = \left(x_\sigma^1, x_\sigma^2, \cdots, x_\sigma^n\right) \in E(v).$$

注意到, 它就是定理 20.6.2 的证明中构造的分配, 它被证明属于核心. 故 $x_\sigma \in C(v)$. 但

$$\varphi(v) = \frac{1}{n!} \sum_{\sigma \in \mathbf{S}_n} x_\sigma$$

是所有 x_σ 的凸组合, 而 $C(v)$ 是凸集, 故 $\varphi(v) \in C(v)$. □

下面的定理给出充要条件.

定理 20.8.2　设 (N, v) 为一合作博弈, 则 $\varphi(v) \in C(v)$ 当且仅当

$$V_v\left(\varXi_n M_n - I_{2^n}\right) \geqslant 0. \tag{20.8.1}$$

这里, $\varXi_n$ 由 (20.7.16) 定义, M_n 由 (20.5.2) 定义.

证明　$\varphi(v)$ 作为分配, 显然满足 (20.5.5). 根据定理 20.5.1, 只要 $\varphi(v)$ 满足 (20.5.4) 即可. 由等式 (20.5.3) 及 (20.7.15) 立得结论. □

例 20.8.1　考察一个 3 人合作博弈 (N, v). 设

$$v(S) = \begin{cases} 1, & S = \{i\},\ i = 1, 2, 3, \\ 4, & S = \{i, j\},\ i \neq j, \\ 6, & S = N, \\ 0, & S = \varnothing. \end{cases}$$

于是有

$$V_v = \begin{bmatrix} 6 & 4 & 4 & 1 & 4 & 1 & 1 & 0 \end{bmatrix}.$$

注意到

$$\varXi_3 = \frac{1}{6} \begin{bmatrix} 2 & 2 & 2 \\ 1 & 1 & -2 \\ 1 & -2 & 1 \\ 2 & -1 & -1 \\ -2 & 1 & 1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \\ -2 & -2 & -2 \end{bmatrix}.$$

则 Shapley 值为

$$\varphi(v) = V_v \Xi_3 = \begin{bmatrix} 2 & 2 & 2 \end{bmatrix}.$$

检验 (20.8.1), 注意到

$$M_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}.$$

易知

$$V_v \left(\Xi_3 M_3 - I_8 \right) = [0\,0\,0\,1\,0\,1\,1\,0] \geqslant 0.$$

因此, 上述 Shapley 值属于核心.

20.9 注释与参考

博弈理论大致可以分为两个部分: 合作博弈 (cooperative game) 与非合作博弈 (non-cooperative game). 前几章主要讨论非合作博弈, 本章讨论合作博弈. 合作博弈与非合作博弈是从不同角度对于博弈思想进行刻画和分析的两种工具, 不能完全分开. 非合作博弈是策略导向的, 它研究博弈的微观层面; 合作博弈是成果导向的, 它研究博弈的宏观层面. 在一个合作博弈中, 特征函数揭示的是联盟可能带来的收益. 而合作博弈的关键是如何在参加者中合理地分配收益. 因此, 寻找合理的分配就是合作博弈的目标.

本章内容可参考文献 [12, 26, 32]. 关于"Shapley 值"计算以及"无异议博弈"部分, 可见文献 [4, 61].

第 21 章　演化博弈的优化控制

21.1　输入-状态转移图

考察一个 k 值逻辑控制系统如下:

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ x_2(t+1) = f_2(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ \quad\vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \end{cases} \tag{21.1.1}$$

这里 $x_i, u_j \in \mathcal{D}_k$, x_i 为状态变量, u_j 为控制变量, f_i 为 k 值逻辑函数. (21.1.1) 简记为

$$X(t+1) = F(X(t), U(t)), \tag{21.1.2}$$

这里 $X = (x_1, \cdots, x_n)$, $U = (u_1, \cdots, u_m)$.

当 $k = 2$, (21.1.1) 变为一个布尔控制网络. 本章只考虑一般的 k. 本章不考虑一个逻辑控制系统的输出, 因此, 将输出方程略去. 利用向量形式, 设 $x_i, u_i \in \Delta_k$, 并记 $x(t) = \ltimes_{i=1}^n x_i(t)$, $u(t) = \ltimes_{i=1}^m u_i(t)$. 则 (21.1.1) 可表示为其代数状态空间形式如下:

$$x(t+1) = Lu(t)x(t), \tag{21.1.3}$$

这里 $x(t) \in \Delta_{k^n}$, $u(t) \in \Delta_{k^m}$, $L \in \mathcal{L}_{k^n \times k^{m+n}}$.

系统在 t 时刻的支付函数为 $P : \mathcal{D}_k^n \times \mathcal{D}_k^m \to \mathbb{R}$ (在向量形式下 (利用相同的函数记号) 有 $P : \Delta_{k^n} \times \Delta_{k^m} \to \mathbb{R}$). 设 $S(t) = (X(t), U(t))$, 或 $s(t) = u(t) \ltimes x(t)$. 我们将平均支付作为性能指标. 设系统从初态 x_0 出发, 在控制 $u(t)$ 作用下, 系统轨线为 $x(t, x_0, u)$ (如不会产生误会, 则简记作 $x(t)$). $x(t, x_0, u)$ 的平均支付定义为

$$J(x(t, x_0, u)) = J(u) = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} P(x(t), u(t)). \tag{21.1.4}$$

最优控制问题定义为: 找出一个控制 $u^*(t)$, 它能极大化性能指标函数 $J(u)$. 即

$$J(u^*) = \max_u J(u). \tag{21.1.5}$$

这个控制称为最优控制.

解决最优控制问题包括以下几个方面: (i) 最优控制 $u^*(t)$ 是否存在? (ii) 如果存在是否唯一? (iii) 如何构造最优控制? 下面我们逐一回答这几个问题.

首先定义 $\mathcal{S}=\mathcal{D}_k^n\times\mathcal{D}_k^m$ 中的一个环 (在向量形式下变为 $\mathcal{S}=\Delta_{k^{m+n}}$ 中的一个环). 下一定义最早见文献 [113].

定义 21.1.1 一个以 $\mathcal{S}$ 为顶点的有向图称为系统 (21.1.1) 的输入-状态转移图, 如果它的边由如下方法确定: 对任何两个顶点 $S_p=(U_p,X_p)\in\mathcal{S}$ 和 $S_q=(U_q,X_q)\in\mathcal{S}$, 存在一条有向边 $\overrightarrow{S_pS_q}$, 当且仅当

$$X_q=F(U_p,X_p).$$

在向量形式下, 顶点可以用 $s_p=(u_p,x_p)=(\delta_{k^m}^{\alpha},\delta_{k^n}^{\beta})$ 来表示. 这种表示可进一步简化为 $s_p=u_px_p=\delta_{k^{m+n}}^{\gamma}\in\Delta_{k^{m+n}}$. 后者的合理性在于乘积形式的 s_p 可以唯一地分解为 (u_p,x_p). 准确地说, 令 $\delta_{k^{m+n}}^{\gamma}=\delta_{k^m}^{\alpha}\delta_{k^n}^{\beta}$. 那么

$$\gamma=(\alpha-1)k^n+\beta,$$

等价地有

$$\alpha=\Big[\frac{\gamma}{k^n}\Big]+1,\quad \beta=\gamma\ (\mathrm{mod}\ k^n).$$

利用代数形式可知, 转移图具有一条有向边 $S_p\ \to\ S_q$(即 $(s_p,s_q)\in E$), 当且仅当

$$x_q=Lu_px_p\quad (\text{或 } x_q=Ls_p).$$

现在假定 $s_p=\delta_{k^{m+n}}^{\xi}$ 及 $s_q=\delta_{k^{m+n}}^{\eta}$, 那么, 边 (s_p,s_q) 也可以表示为

$$\delta_{k^{m+n}}[\xi,\eta].$$

输入-状态转移图的拓扑结构在网络的优化控制问题上发挥着重大作用.

定义 21.1.2 1) 状态 $s^{'}\in\mathcal{S}$ 称为从 $s=(u,x)\in\mathcal{S}$ 可到达, 如果存在一条由有向边组成的路径, 它从 s 出发且终止于 $s^{'}$. s 的可达集记为 $R(s)$.

2) 输入-状态转移图称为强联通的, 如果对任一 $s\in\mathcal{S}$,

$$R(s)=\mathcal{S}.$$

我们用一个例子解释输入-状态转移图.

例 21.1.1 考察下面这个布尔网络:

$$\begin{cases}x_1(t+1)=(x_1(t)\wedge u(t))\vee x_2(t),\\ x_2(t+1)=(u(t)\wedge x_1(t)\wedge\neg x_2(t))\vee(\neg u(t)\wedge x_1(t)).\end{cases}\tag{21.1.6}$$

利用向量表示, 上述网络的代数形式为

$$x(t+1) = Lu(t)x(t), \tag{21.1.7}$$

这里

$$L = \delta_4[2\ 1\ 2\ 4\ 1\ 3\ 2\ 4].$$

其输入-状态转移图有 8 个顶点, 分别为

$$A = (\delta_2^1, \delta_4^1), \quad B = (\delta_2^1, \delta_4^2), \quad C = (\delta_2^1, \delta_4^3), \quad D = (\delta_2^1, \delta_4^4),$$
$$E = (\delta_2^2, \delta_4^1), \quad F = (\delta_2^2, \delta_4^2), \quad G = (\delta_2^1, \delta_4^3), \quad H = (\delta_2^1, \delta_4^4).$$

根据定义 21.1.1, 不难确定该图的边. 于是, 可以得到其输入-状态转移图如图 21.1.1.

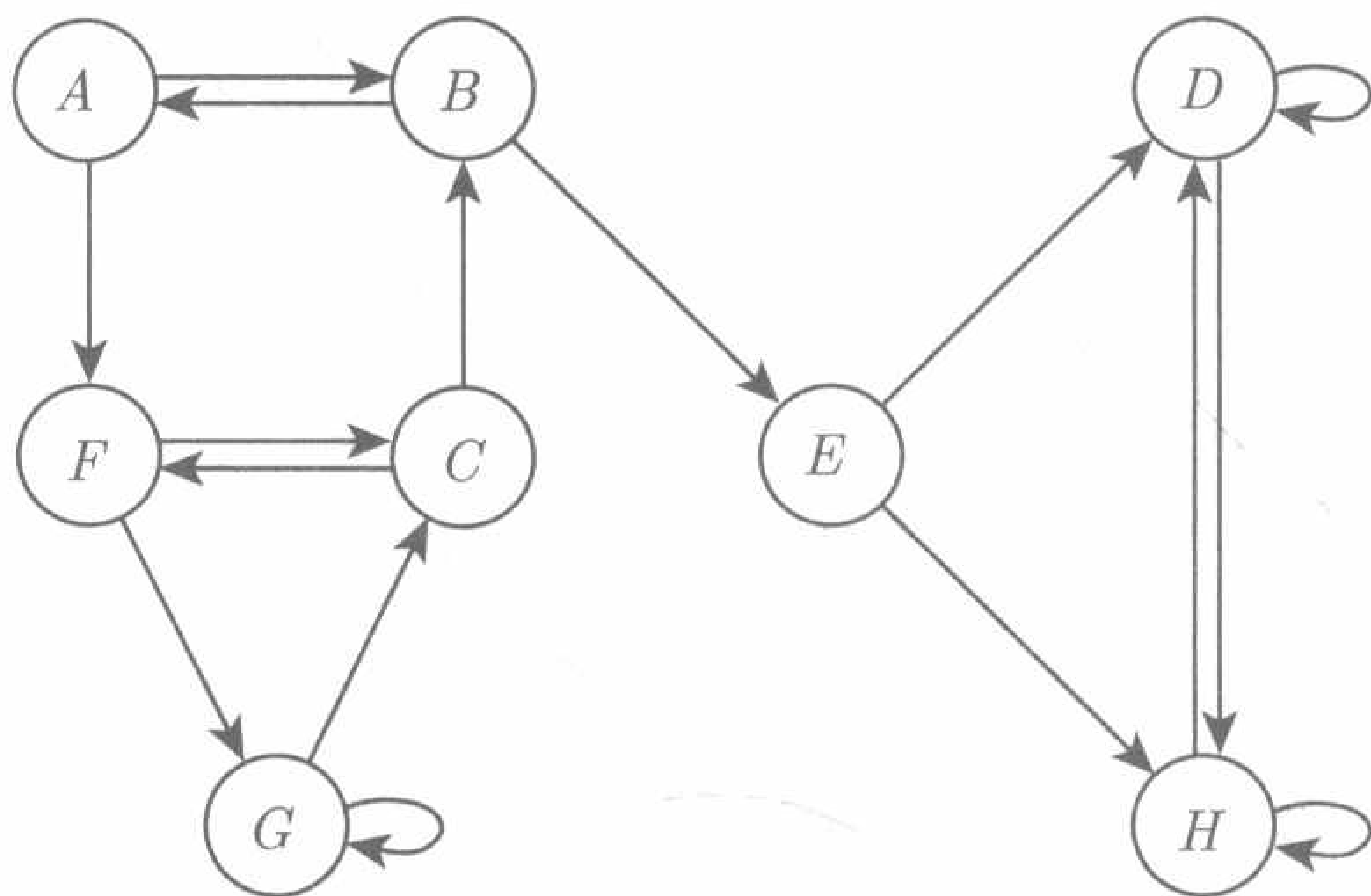


图 21.1.1　例 21.1.1 的输入-状态转移图

从图 21.1.1, 容易看出

$$R(A) = R(B) = R(C) = R(F) = R(G) = \mathcal{S}.$$

此外,

$$R(E) = \{D, H\}, \quad R(H) = \{H, D\}, \quad R(D) = \{D, H\},$$

因此, 该输入-状态转移图不是强联通的.

不难看出, 输入-状态转移图强联通意味着系统完全能控.

定义 21.1.3　1) 设 $s_i \in \mathcal{S}$, $i = 1, 2, \cdots, \ell$. $(s_1, s_2, \cdots, s_\ell)$ 称为输入-状态转移图的一条路径, 如果 (s_i, s_{i+1}), $i = 1, \cdots, \ell - 1$ 是该图的一条边.

2) 一条路径 $(s_1, s_2, \cdots, s_\ell)$ 称为一个环, 如果 $s_{\ell+1} = s_1$. 最小的 $\ell > 0$ 称为环的长度. 长度 $\ell = 1$ 的环称为不动点.

3) 设 $C = (s_1, s_2, \cdots, s_\ell)$ 为一个环并记 $s_i = (u_i, x_i), i = 1, \cdots, \ell$. 如果 $x_i \neq x_j, 1 \leqslant i < j \leqslant \ell$, 那么, C 称为一个单环.

例 21.1.2 考察例 21.1.1 中的输入-状态转移图. 容易看出, 该图有如下这些环: (D), (H), (G), (H,D), (A,B), (F,C), (F,G,C), (A,F,C,B), (A,F,G,C,B) 等.

设 $C:=(s_1,\cdots,s_\ell)\in\mathcal{S}$ 为一环. 则环上的平均支付为

$$P_a(C)=\frac{P(s_1)+\cdots+P(s_\ell)}{\ell}.$$

于是可知如下结果[113]

命题 21.1.1 1) 设 $S:=(s_1,\cdots,s_T)$ 为系统 (21.1.1) 的输入-状态转移图的一路径. 那么

$$S=\bigcup_{i=1}^{N}C_i\cup R, \tag{21.1.8}$$

这里 C_i 是某些环, R 是未能组成环的剩余元素, 因此, $|R|\leqslant k^{m+n}$.

2) 设 $S:=(s_1,\cdots,s_T)$ 为系统 (21.1.1) 的输入-状态转移图的一路径, $\mathcal{C}$ 为其环集合, 它是有限集. 设 $C^*\in\mathcal{C}$, 使得

$$P_a(C^*)=\max\{P_a(C),\ \forall C\in\mathcal{C}\}.$$

那么

$$J(S)\leqslant P_a(C^*). \tag{21.1.9}$$

证明 1) 将 S 中的环一个一个移走. 最后剩下的至多为 k^{m+n} 元素, 因为 $|\mathcal{S}|=k^{m+n}$. 注意 (21.1.8) 指的是在元素集合意义下的环, 即, 我们无需担心: 是否一个环的元素真正连结而形成一个环. 这就证明了第一个断言.

2) 设 S 分解为一些长度为 ℓ_i 的环 C_i, $i=1,\cdots,N$ 以及剩余 R. 那么

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}P(S_t)&=\sum_{i=1}^{N}\left[\frac{\ell_1}{T}P_a(C_1)+\cdots+\frac{\ell_N}{T}P_a(C_N)\right]+\frac{P(r_1)+\cdots+P(r_q)}{T}\\&\leqslant\frac{\sum_{i=1}^{N}\ell_i}{T}P_a(C^*)+\frac{P(r_1)+\cdots+P(r_q)}{T}\\&=\frac{T-q}{T}P_a(C^*)+\frac{P(r_1)+\cdots+P(r_q)}{T},\end{aligned} \tag{21.1.10}$$

这里 $\{r_1,\cdots,r_q\}=R$, $q\leqslant k^{m+n}$. 因此,

$$\lim_{T\to\infty}\frac{T-q}{T}P_a(C^*)+\frac{P(r_1)+\cdots+P(r_q)}{T}=P_a(C^*),$$

结论显见. □

我们定义状态 x_0 的可达集如下:

$$R(x_0) = \bigcup_{u_0 \in \mathcal{D}_k^m} R(s_0 = (u_0, x_0));$$

该可达集中的环定义为

$$\mathcal{C}_{x_0} = \{C \in \mathcal{C} \mid C \subset R(x_0)\};$$

其中的最优环定义为: $C_{x_0}^* \in \mathcal{C}_{x_0}$ 满足

$$P_a(C_{x_0}^*) \geqslant P_a(C), \quad \forall C \in \mathcal{C}_{x_0}.$$

那么, 显然有如下结果

推论 21.1.1 考察网络 (21.1.1) 在性能指标 (21.1.4) 下的最优控制问题. 最优控制使轨道收敛到 $C_{x_0}^*$, 且性能指标的最优值为 $J_{\max} = P_a(C_{x_0}^*)$. 如果 $C^* \subset R(x_0) = \mathcal{S}$, 则性能指标的最优值为 $J_{\max} = P_a(C^*)$. 如果输入 - 状态转移图是强联通的, 那么, 最优性能指标值为 $P_a(C^*)$, 它与系统出发点无关.

21.2 逻辑控制网络的拓扑结构

要考虑逻辑控制系统的优化问题, 就必须先考察它的输入-状态转移图的拓扑结构. 特别是, 由命题 21.1.1 可知, 最优轨线会收敛于某个极限环, 因此, 计算极限环成为关键. 在以下讨论中, 假定输入-状态转移图是强联通的. 否则, 只考虑从初始点 x_0 出发的可达集 $R(x_0)$ 内的情况 (可达集内的环与最优解).

一个 k 值逻辑控制网络可表示为 (21.1.1), 那里 $x_i, u_i \in \mathcal{D}_k$. 其代数状态空间表示形式为 (21.1.3), 那里 $L \in \mathcal{L}_{k^n \times k^{m+n}}$, $x = \ltimes_{i=1}^n x_i, u = \ltimes_{j=1}^m u_j$, 其中 $x_i, u_i \in \Delta_k$.

我们需要考察控制-状态空间 $\mathcal{S}$ 中的环. 记 $\mathrm{Col}_i(L)$ 为矩阵 L 的第 i 列. 那么, 可以证明如下结论:

命题 21.2.1 一条边 $\delta_{k^{m+n}}(i,j)$ 存在, 当且仅当

$$\mathrm{Col}_i(L) = \delta_{k^n}^{\ell}, \quad 这里 \quad \ell = j \pmod{k^n}. \tag{21.2.1}$$

证明 根据定义, 边 $\delta_{k^{m+n}}(i,j)$ 存在当且仅当存在控制 $u(t+1)$ 使得

$$u(t+1)L\delta_{k^{m+n}}^i = \delta_{k^{m+n}}^j. \tag{21.2.2}$$

显见 $L\delta_{k^{m+n}}^i = \mathrm{Col}_i(L)$, 因此, 由 (21.1.3) 可得

$$u(t+1)\mathrm{Col}_i(L) = \delta_{k^{m+n}}^j. \tag{21.2.3}$$

注意到 $\delta_{k^{m+n}}^j$ 可唯一地分解为 $\delta_{k^m}^{\xi}\delta_{k^n}^{\ell}$, 这里 $j = (\xi-1)k^n + \ell$, 结论显见. □

例 21.2.1 设一布尔控制网络为

$$x(t+1) = Lu(t)x(t), \tag{21.2.4}$$

这里 $u(t), x(t) \in \Delta$, 且

$$L = \delta_2[1\ 2\ 2\ 1].$$

注意到 $\delta_4^1 \sim (1,1)$, $\delta_4^2 \sim (1,0)$, $\delta_4^3 \sim (0,1)$, $\delta_4^4 \sim (0,0)$, 那么, 可以得到图 21.2.1.

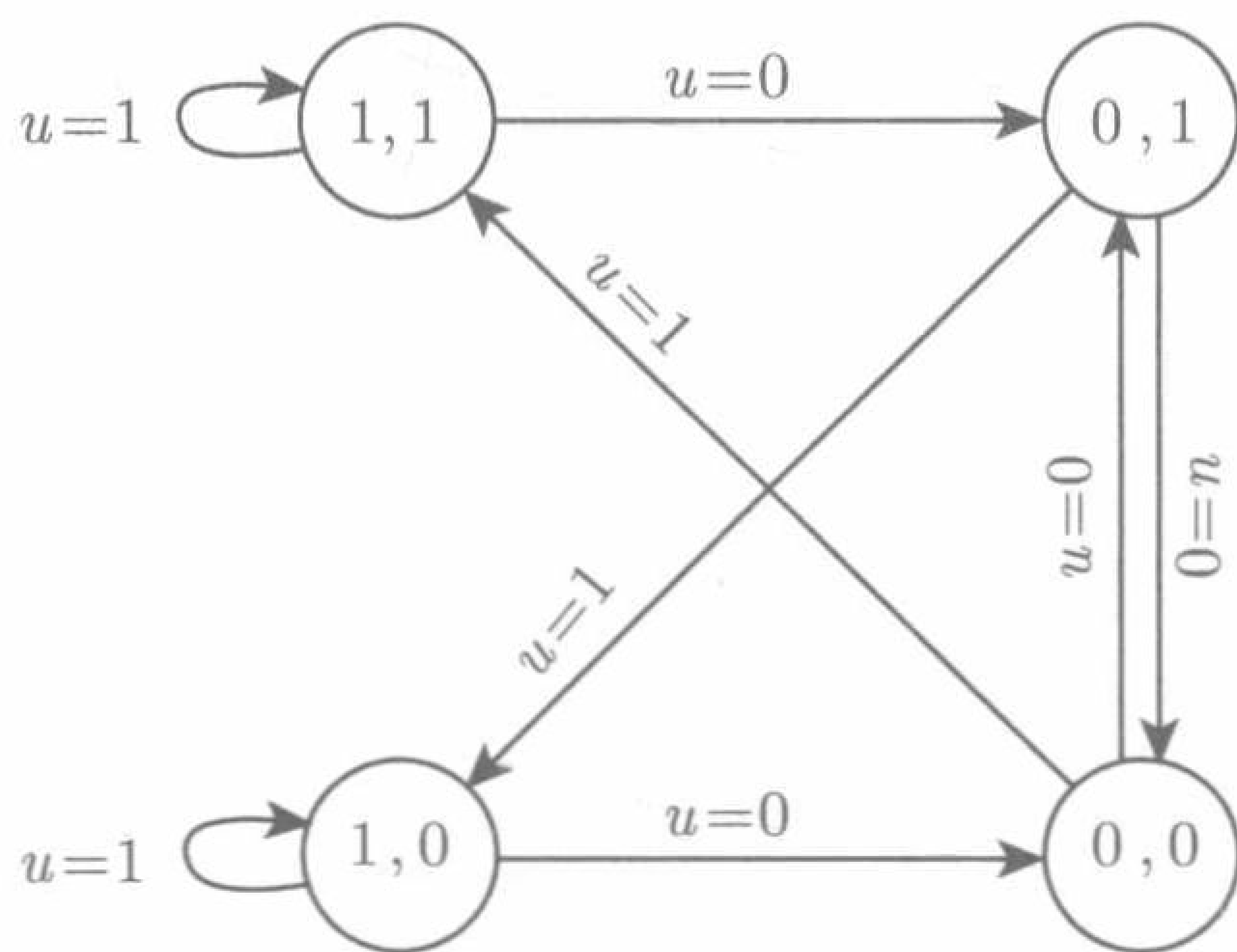


图 21.2.1 输入-状态转移图解

由图 21.2.1 可以看出: (1,1) 和 (1,0) 为不动点, 且系统具有如下长度不超过 4 的极限环:

$$\begin{array}{ll} (0,1) \to (0,0); & (0,1) \to (1,0) \to (0,0); \\ (1,1) \to (0,1) \to (0,0); & (0,0) \to (1,1) \to (0,1) \to (1,0); \\ (1,1) \to (1,1) \to (0,1) \to (0,0); & (1,0) \to (1,0) \to (0,0) \to (0,1). \end{array}$$

在一些简单情况下, 不动点与极限环可以从输入-状态转移图直接得到. 但当 m 和 n 较大时, 很难像上例那么直接从图中得到. 所以我们需要发现公式, 然后通过代数计算求得它们.

由 (21.1.3) 可知:

$$\begin{aligned} x(t+d) &= Lu(t+d-1)x(t+d-1) \\ &= Lu(t+d-1)Lu(t+d-2)\cdots Lu(t+1)Lu(t)x(t) \\ &= L(I_{k^m} \otimes L)u(t+d-1)u(t+d-2)Lu(t+d-3) \\ &\quad \cdot Lu(t+d-4)\cdots Lu(t)x(t) \\ &:= L_d(\ltimes_{\ell=1}^d u(t+d-\ell))x(t), \end{aligned} \tag{21.2.5}$$

这里

$$L_d = \prod_{i=1}^d (I_{k^{(i-1)m}} \otimes L) \in \mathcal{L}_{k^n \times k^{dm+n}}. \tag{21.2.6}$$

在计算极限环之前, 引进一些记号.

- 设 $d \in \mathbb{Z}_+$, 记 $\mathcal{P}(d)$ 为 d 的真因子集.
- 设 $i, k, m \in \mathbb{Z}_+$, 那么

$$\theta_k^m(d,i) := \{(\ell, j) \mid \exists \ell \in \mathcal{P}(d) \text{ 和 } j, \text{ 使得 } \delta_{k^{dm}}^i = (\delta_{k^{\ell m}}^j)^{\frac{d}{\ell}}\}. \tag{21.2.7}$$

下面的例子用来解释如上的记号.

例 21.2.2 1) 设 $d=6$. 则 $\mathcal{P}(d)=\{1,2,3\}$.

2) 设 $m,k,d \in \mathbb{Z}_+$ 给定. 那么, 根据显见等式 $\delta_{k^\alpha}^a \delta_{k^\beta}^b = \delta_{k^{\alpha+\beta}}^{(a-1)k^\beta+b}$, 对每一个 $\ell \in \mathcal{P}(d)$ 至少存在一个 j, 使得 $(\ell, j) \in \theta_k^m(i,d)$.

例如, $m=k=2$, $d=6$.

- 设 $i=1$. 则
  对 $\ell=1$, $\delta_{k^{dm}}^i = \delta_{2^{12}}^1 = (\delta_{2^2}^1)^6 = (\delta_{k^{\ell m}}^j)^{\frac{d}{\ell}}$, 因此 $j=1$;
  对 $\ell=2$, $\delta_{2^{12}}^1 = (\delta_{2^4}^1)^3$, 因此 $j=1$;
  对 $\ell=3$, $\delta_{2^{12}}^1 = (\delta_{2^6}^1)^2$, 因此 $j=1$.
  于是有, $\theta_2^2(6,1)=\{(1,1),(2,1),(3,1)\}$.
- 设 $i=2$, 对任何 $\ell \in \mathcal{P}(6)$ 及任何 $1 \leqslant j \leqslant 2^{2\ell}$, $\delta_{2^{12}}^2 \neq (\delta_{2^{2\ell}}^j)^{\frac{d}{\ell}}$, 因此无解.
  即, $\theta_2^2(6,2)=\varnothing$.
- 设 $i=2^6+2$,
  当 $\ell=1$ 或 2, 易证无解;
  设 $\ell=3$, $\delta_{2^{12}}^{2^6+2} = (\delta_{2^6}^2)^2$, 故 $j=2$.
  因此, $\theta_2^2(6,2^6+2)=\{(3,2)\}$.

以下简记 $\theta_k^m(d,i)$ 为 $\theta(d,i)$, 默认的 k 和 m 分别为逻辑类型 (逻辑变量取值个数) 以及网络输入 (控制) 个数.

记 $\mathrm{Blk}_i(L)$ 为 $n \times nm$ 的逻辑矩阵 L 的第 i 个 $n \times n$ 块. 那么有

定理 21.2.1 在 k 值逻辑控制网络 (21.1.3) 的输入-状态转移图中, 长度为 d 的极限环的个数可由下式递推算出:

$$N_d = \frac{1}{d}\sum_{i=1}^{k^{dm}} T(\mathrm{Blk}_i(L_d)), \tag{21.2.8}$$

这里 L_d 由 (21.2.6) 定义, 且

$$T(\mathrm{Blk}_i(L_d)) = \mathrm{Trace}(\mathrm{Blk}_i(L_d)) - \sum_{(\ell,j)\in\theta(d,i)} T(\mathrm{Blk}_j(L_\ell)). \tag{21.2.9}$$

证明 $\mathcal{S}$ 中的每个环是一个状态和控制空间的乘积环. 我们先看状态空间的环:

设 $x(t)$ 属于状态空间一个长度为 d 的环, 从 (21.2.5) 可知

$$x(t) = L_d(\ltimes_{\ell=1}^{d} u(t+d-\ell))x(t).$$

如果 $u(t+d-1),\cdots,u(t)$ 固定, 例如 $\ltimes_{\ell=1}^{d} u(t+d-\ell) = \delta_{k^{dm}}^{i}$, 那么

$$x(t) = \mathrm{Blk}_i(L_d)x(t).$$

如果 $x(t) = \delta_{k^n}^{j}$, 即 $\mathrm{Blk}_i(L_d)$ 的 (j,j) 位元素为 1. 于是在状态空间及控制 $u(t+d-1),\cdots,u(t)$ 作用下长度为 d 的环为

$$\{x(t), Lu(t)x(t), L_2u(t+1)u(t)x(t),\cdots, L_du(t+d-1)\cdots u(t)x(t)\}.$$

因此, 将该环与给定控制相乘, 就得到一个在控制-状态乘积空间的长度为 d 的环. 可见, 长度为 d 的环的个数 (包括重复循环多次的环) 为 $\dfrac{1}{d}\sum_{i=1}^{k^{dm}}\mathrm{Trace}(\mathrm{Blk}_i(L_d))$.

注意到如果 ℓ 是 d 的真因子, 而设 $x(t)$ 在控制 $\widetilde{u}(t+\ell-1)\cdots\widetilde{u}(t) = \delta_{k^{\ell m}}^{j}$ 作用下长度为 ℓ 的环上, 同时也在控制 $u(t+d-1)\cdots u(t) = \delta_{k^{dm}}^{i}$ 作用下长度为 d 的环上, 这两个环在输入 - 状态转移图中为同一个环, 当且仅当 $\delta_{k^{dm}}^{i} = (\delta_{k^{\ell m}}^{j})^{\frac{d}{\ell}}$. 除去重复循环多次的环, 即可得到 (21.2.8). □

值得说明的是, (21.2.9) 依 d 递归地定义了所有的 $T(\mathrm{Blk}_i(L_d))$. 注意到当 $d=1$ 时,

$$\theta(i,1) = \phi, \quad \forall i.$$

因此 $T(\mathrm{Blk}_i(L_d))$ 当 $d=1$ 时是定义好的. 于是, 当 $d>1$ 时, 它们可以递归地定义.

对于长度为 d 的环 C, 由于 $s(t) = \delta_{k^{m+n}}^{\ell}$ 可唯一地分解为 $u(t)x(t) = \delta_{k^m}^{i}\delta_{k^n}^{j}$, 环 C 可确定为

$$C = \left((\delta_{k^m}^{i(t)}, \delta_{k^n}^{j(t)}), (\delta_{k^m}^{i(t+1)}, \delta_{k^n}^{j(t+1)}), \cdots, (\delta_{k^m}^{i(t+d-1)}, \delta_{k^n}^{j(t+d-1)})\right).$$

为简洁, 将它记为

$$C = \delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1))). \tag{21.2.10}$$

单环曾在定义 21.1.3 中给出. 这里给出一个等价定义: 一个环 $C = \delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1)))$ 称为单环, 如果它满足

$$i(\xi) \neq i(\ell), \quad t \leqslant \xi < \ell \leqslant t+d-1. \tag{21.2.11}$$

例 21.2.3　回忆例 21.2.1. 因为

$$L_1 = L = \delta_2[1\ 2\ 2\ 1],$$

可知 $\mathrm{Trace}(\mathrm{Blk}_1(L_1)) = 2$, $\mathrm{Trace}(\mathrm{Blk}_2(L_1)) = 0$. 因此, δ_2^1 和 δ_2^2 为在控制 $u = \delta_2^1$ 下的不动点. 所以 $N_1 = 1$, 并且在输入-状态转移图中的不动点为

$$\delta_2 \times \delta_2((1,1)), \quad \delta_2 \times \delta_2((1,2)),$$

它们都是单环. 其次, 因为

$$L_2 = L(I_2 \otimes L) = \delta_2[1\ 2\ 2\ 1\ 2\ 1\ 1\ 2],$$

我们有 $\mathrm{Trace}(\mathrm{Blk}_1(L_2)) = \mathrm{Trace}(\mathrm{Blk}_4(L_2)) = 2$, $\mathrm{Trace}(\mathrm{Blk}_2(L_2)) = \mathrm{Trace}(\mathrm{Blk}_3(L_2)) = 0$, $\delta_4^1 = \delta_2^1\delta_2^1$, $\delta_4^4 = \delta_2^2\delta_2^2$, 因此,

$$\begin{aligned}
T(\mathrm{Blk}_1(L_2)) &= \mathrm{Trace}(\mathrm{Blk}_1(L_2)) - T(\mathrm{Blk}_1(L_1)) = 0,\\
T(\mathrm{Blk}_4(L_2)) &= \mathrm{Trace}(\mathrm{Blk}_4(L_2)) - T(\mathrm{Blk}_2(L_1)) = 2,
\end{aligned}$$

$T(\mathrm{Blk}_2(L_2)) = T(\mathrm{Blk}_3(L_2)) = 0$, 因而, $N_2 = 1$. δ_2^1 和 δ_2^2 处于在控制 $u(t+1)u(t) = \delta_2^2\delta_2^2$ 下长度为 2 的环上. 这样可得到在输入-状态转移图中长度为 2 的环

$$\delta_2 \times \delta_2((2,1),(2,2)),$$

它是一个单环. 考虑

$$L_3 = L(I_2 \otimes L)(I_4 \otimes L) = \delta_2[1\ 2\ 2\ 1\ 2\ 1\ 1\ 2\ 1\ 2\ 2\ 1].$$

由于 $\mathrm{Trace}(\mathrm{Blk}_1(L_3)) = \mathrm{Trace}(\mathrm{Blk}_4(L_3)) = \mathrm{Trace}(\mathrm{Blk}_6(L_3)) = \mathrm{Trace}(\mathrm{Blk}_7(L_3)) = 2$, 故

$$T(\mathrm{Blk}_4(L_3)) = T(\mathrm{Blk}_6(L_3)) = T(\mathrm{Blk}_7(L_3)) = 2, T(\mathrm{Blk}_i(L_3)) = 0, i = 1,2,3,5,8,$$

于是, $N_3 = 2$. δ_2^1 和 δ_2^2 处于在控制 $u(t+2)u(t+1)u(t) = \delta_8^4 = \delta_2^1\delta_2^2\delta_2^2$ 下的长度为 3 的环上, $\delta_8^6 = \delta_2^2\delta_2^1\delta_2^2$, 及 $\delta_8^7 = \delta_2^2\delta_2^2\delta_2^1$. 这样可得到在输入-状态转移图中长度为 3 的环

$$\delta_2 \times \delta_2((1,1),(2,1),(2,2)), \quad \delta_2 \times \delta_2((2,1),(1,2),(2,2)).$$

最后, 因为

$$\begin{aligned}
L_4 &= L(I_2 \otimes L)(I_4 \otimes L)(I_8 \otimes L)\\
&= \delta_2[1\ \ 2\ \ 2\ \ 1\ \ 2\ \ 1\ \ 1\ \ 2\ \ 2\ \ 1\ \ 1\ \ 2\ \ 1\ \ 2\ \ 2\ \ 1\\
&\qquad 2\ \ 1\ \ 1\ \ 2\ \ 1\ \ 2\ \ 2\ \ 1\ \ 1\ \ 2\ \ 2\ \ 1\ \ 2\ \ 1\ \ 1\ \ 2],
\end{aligned}$$

有 $\text{Trace}(\text{Blk}_i(L_4)) = 2, i = 1,4,6,7,10,11,13,16$.

因此,

$$T(\text{Blk}_i(L_4)) = \begin{cases} 2, & i = 4,6,7,10,11,13, \\ 0, & \text{其他}. \end{cases}$$

于是 $N_4 = 3$. δ_2^1 及 δ_2^2 为在控制 $u(t+3)u(t+2)u(t+1)u(t) = \delta_{16}^4 = \delta_2^1\delta_2^1\delta_2^2\delta_2^2$ 下长度为 4 的环上, 而 $\delta_{16}^6 = \delta_2^1\delta_2^2\delta_2^1\delta_2^2$, $\delta_{16}^7 = \delta_2^1\delta_2^2\delta_2^2\delta_2^1$, $\delta_{16}^{10} = \delta_2^2\delta_2^1\delta_2^1\delta_2^2$, $\delta_{16}^{11} = \delta_2^2\delta_2^1\delta_2^2\delta_2^1$, 及 $\delta_{16}^{13} = \delta_2^2\delta_2^2\delta_2^1\delta_2^1$. 这样可得到在输入-状态转移图中长度为 4 的环

$$\begin{aligned} &\delta_2 \times \delta_2((1,1),(2,1),(1,2),(2,2)), \\ &\delta_2 \times \delta_2((1,2),(1,2),(2,2),(2,1)), \\ &\delta_2 \times \delta_2((1,1),(1,1),(2,1),(2,2)). \end{aligned}$$

这个结果与我们在例 21.2.1 中得到的结果一致.

21.3 逻辑控制系统的最优控制

本节考虑逻辑动态网络的最优控制及相应的最优轨道问题.

考察系统 (21.1.1), 设其指标函数为

$$J(u) = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} P(x(t), u(t)), \tag{21.3.1}$$

这里 $P: \mathcal{D}_k^{n+m} \to \mathbb{R}$ 为伪逻辑函数. 我们的目标是找到最优控制 $u^*(t)$ 以极大化 $J(u)$, 即

$$J^*(u) := J(u^*) = \max_u J(u).$$

类似文献 [113], 可以证明以下结果:

定理 21.3.1 *考虑 k 值控制网络 (21.1.1) 以及指标函数 (21.3.1), 存在最优控制 $u^*(t)$ 它可以极大化指标函数. 同时, 在有限步后最优轨线 $s^*(t) = u^*(t)x^*(t)$ 将变为周期映射.*

定理 21.3.1 将文献 [113] 中对单变量布尔网络的结果推广到多变量 k 值逻辑网络. 但是, 当把问题转化为图时, 它们没有本质的区别, 因此, 证明是类似的. 定理 21.3.1 保证最优控制能在某条轨线上实现, 而该轨线收敛于一个环. 下面, 利用逻辑函数的矩阵表示, 提出一种方法寻找最优轨线. 我们找到一个矩阵 G^*, 称为最优反馈控制矩阵, 使得

$$u^*(t+1) = G^* u^*(t) x^*(t).$$

命题 21.3.1 极限值

$$J(u^*) := \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} P(x^*(t), u^*(t)) \tag{21.3.2}$$

总存在.

证明 考察系统 (21.1.1). 根据定理 21.3.1, 一条最优轨线将收敛于一个吸引子 (不动点或极限环). 作为极限, $J(u^*)$ 变为吸引子上的平均值. □

命题 21.3.2 对一个环 C, 总存在一个单环 C_s, 使得

$$P_a(C_s) \geqslant P_a(C). \tag{21.3.3}$$

证明 记 $C = \delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1)))$ 为一个任意的环. 如果它是单环, 命题显见. 否则, 设 $\delta_{k^n}^{j(\xi)} = \delta_{k^n}^{j(\ell)}$, $\xi < \ell$, 以及 $C_1 = \delta_{k^m}\times\delta_{k^n}((i(\xi),j(\xi)),\cdots,(i(\ell-1),j(\ell-1)))$ 为一单环. 如果 $P_a(C_1)\geqslant P_a(C)$, 命题显见.

否则, 移去 C_1, 剩余部分形成一个新的环 C_1', 由于 $L\delta_{k^m}^{i(\xi-1)}\delta_{k^n}^{j(\xi-1)} = \delta_{k^n}^{i(\xi)} = \delta_{k^n}^{i(\ell)}$, 故有 $P_a(C_1') > P_a(C)$. 如果 C_1' 是单环, 命题成立. 否则, 可以找到 C_2, 使得或者 (21.3.3) 成立, 或者移走它. 继续这个过程, 直到最后可以找到一个单环 C_s, 使得 (21.3.3) 成立. □

记 $R(x)$ 为状态 x 的可达集, 称 $C\subset R(x)$, 如果 C 的每一个元素都属于 $R(x)$.

定义 21.3.1 给定初始值 x_0, 一个环 C^* 称为最优环, 如果它满足

$$C^* \in \underset{C\subset R(x_0)}{\operatorname{argmax}}\, P(C). \tag{21.3.4}$$

根据 (21.2.5), 在第 d 步轨线可以从 x_0 到达

$$R_d(x_0) = \{u(d)L_d \ltimes_{\ell=1}^{d} u(d-\ell)x_0 \mid \forall\, u(\ell)\in\Delta_{k^m}, 0\leqslant \ell\leqslant d\}.$$

如果 $x_0 = \delta_{k^n}^{j(0)}$,

$$R_d(x_0) = \{u(d)\mathrm{Col}_\ell(L_d) \mid \forall\, u(d)\in\Delta_{k^m}, \ell = j(0) \ (\mathrm{mod}\ k^n)\}.$$

如果 $\delta_{k^m}^i\delta_{k^n}^j$ 可由 x_0 出发经 d 步到达, $d\geqslant k^n$, 则从起点到达 $\delta_{k^m}^i\delta_{k^n}^j$ 的轨线至少会通过某个点两次. 类似命题 21.3.2 的证明可知, 从 x_0 出发, 状态 $\delta_{k^m}^i\delta_{k^n}^j$ 可在第 d' 步到达, 这里, $d' < k^n$. 因此,

$$R(x_0) = \bigcup_{d=0}^{k^n-1} R_d(x_0). \tag{21.3.5}$$

根据以上讨论, 要寻找最优环 C^*, 我们仅在包括于 $R(x_0)$ 的所有单环中找就行了. 记从起点到 C^* 的最短距离轨线为

$$(\delta_{k^m}^{i(0)}\delta_{k^n}^{j(0)},\delta_{k^m}^{i(1)}\delta_{k^n}^{j(1)},\cdots,\delta_{k^m}^{i(T_0-1)}\delta_{k^n}^{j(T_0-1)}), \tag{21.3.6}$$

这里

$$C^*=\delta_{k^m}\times\delta_{k^n}((i(T_0),j(T_0)),\cdots,(i(T_0+d-1),j(T_0+d-1))).$$

称 (21.3.6) 为最优轨道.

下面证明存在最优矩阵 G^*.

定理 21.3.2 考察 k 值逻辑控制网络 (21.1.1) 以及目标函数 (21.1.4). 设最优轨线为 (21.3.6), 最优控制为 $u^*(t)$. 则存在一个逻辑矩阵 $G^*\in\mathcal{L}_{k^n\times k^{n+m}}$, 满足

$$\begin{cases}x^*(t+1)=Lu^*(t)x^*(t),\\ u^*(t+1)=G^*u^*(t)x^*(t).\end{cases} \tag{21.3.7}$$

证明 根据命题 21.3.2, 我们可以仅从单环集合中找到最优环. 由于一个单环长度不可能超过 k^n, 设一轨线初态为 $\delta_{k^n}^{j(0)}$, 我们可以找到所有长度小于或等于 k^n 的环, 它们可以从初态到达, 再从中找出最佳轨线 (21.3.6). 易知, $T_0+d\leqslant k^{m+n}$, 因此可以找到最优控制矩阵 G^* 的 T_0+d 列, 满足

$$\mathrm{Col}_s(G^*)=\begin{cases}\delta_{k^m}^{i(\ell+1)}, & s=(i(\ell)-1)k^n+j(\ell),\ell\leqslant T_0+d-2,\\ \delta_{k^m}^{i(T_0)}, & s=(i(T_0+d-1)-1)k^n+j(T_0+d-1),\end{cases} \tag{21.3.8}$$

而 G^* 的其余列可任选. (注意, $\mathrm{Col}(G^*)\subset\Delta_{k^m}$.) 这样就构造了 G^*. □

例 21.3.1 回忆例 21.2.1 和例 21.2.3.

令

$$P(u(t),x(t))=u^{\mathrm{T}}(t)\begin{bmatrix}1&2\\3&4\end{bmatrix}x(t).$$

设初始状态为 $x_0=\delta_2^2$, 由例 21.2.3 的结果可以看出 $C^*=\delta_2\times\delta_2((2,1),(2,2))$ 显然是最优环. 选 $u(0)=\delta_2^2$, 则最优环和从 δ_2^2 到达该环的最短路径为

$$\delta_2\times\delta_2((2,2),(2,1)).$$

因此, $G^*=\delta_2[i\ j\ 2\ 2]$, 其中 i, j 可任选为 1 或 2.

例 21.3.2 考察如下无穷重复博弈. 两玩家 1 和 2 均有 3 个策略: $\{L, M, R\}$. 支付双矩阵见表 21.3.1.

表 21.3.1　支付双矩阵

$P_1\backslash P_2$	L	M	R
L	$3,3$	$0,4$	$9,2$
M	$4,0$	$4,4$	$5,3$
R	$2,9$	$3,5$	$6,6$

容易看出 (M,M) 是一步博弈的唯一纳什均衡点, 这里 $(M,\ M)$ 指两玩家均选 M. 但显然, (R,R) 比 (M,M) 更有效. 在无穷重复的博弈中, 设第 2 个玩家的策略规定如下: 他首轮出 R, 考虑他在第 t 步的策略: 如果在 $(t-1)$ 步的策略是 (R,R) 则出 R, 否则出 M. 这个策略称为 "一报还一报"[79].

设 $L\sim 1,\ M\sim 0.5,\ R\sim 0$. 上述博弈可写成

$$x(t+1)=Lu(t)x(t), \tag{21.3.9}$$

这里

$$L=\delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3],$$

$x(t)\in \Delta_3$, 为状态, 是第 2 玩家在时间 t 的策略; $u(t)\in\Delta_3$, 是控制, 是第 1 玩家在时间 t 的策略.

一个熟知的事实是, 如果支付函数为

$$(1-\delta)\sum_{t=1}^{\infty}\delta^{t-1}\pi_t,$$

这里 π_t 是在时间 t 的支付, δ 是折扣因子, 当 δ 足够接近 1 时, 一报还一报是无限重复博弈的一个纳什均衡点[79].

忽略折扣因子 (即设 $\delta=1$), 则玩家 1 和 2 的支付函数为

$$J_1=\overline{\lim_{T\to\infty}}\frac{1}{T}\sum_{t=1}^{T}P_1(x(t),u(t)),$$

$$J_2=\overline{\lim_{T\to\infty}}\frac{1}{T}\sum_{t=1}^{T}P_2(x(t),u(t)),$$

这里

$$P_1(x(t),u(t))=u^{\mathrm{T}}(t)\begin{bmatrix}3&0&9\\4&4&5\\2&3&6\end{bmatrix}x(t),$$

$$P_2(x(t),u(t))=u^{\mathrm{T}}(t)\begin{bmatrix}3&4&2\\0&4&3\\9&5&6\end{bmatrix}x(t).$$

一个自然的问题是: 此时一报还一报还是其纳什均衡点吗?

设玩家 2 采用一报还一报, 我们讨论玩家 1 的最佳策略. 这时, 问题变为考虑 3 值逻辑控制网络 (21.3.9) 的最优控制, 使之最大化 J_1.

我们先计算输入-状态转移图的环. 首先,

$$L_1 = L = \delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3],$$

因此有 $\mathrm{Trace}(\mathrm{Blk}_1(L_1)) = 1$, $\mathrm{Trace}(\mathrm{Blk}_2(L_1)) = 1$, $\mathrm{Trace}(\mathrm{Blk}_3(L_1)) = 2$, $N_1 = 4$. δ_3^2 是在控制 $u = \delta_3^i, i = 1, 2, 3$ 下的不动点, 而 δ_3^3 是在控制 $u = \delta_3^3$ 下的不动点. 于是系统 (21.3.9) 的不动点为

$$\delta_3 \times \delta_3((1,2)),\ \delta_3 \times \delta_3((2,2)),\ \delta_3 \times \delta_3((3,2)),\ 以及\ \delta_3 \times \delta_3((3,3)),$$

$$L_2 = L(I_3 \otimes L) = \delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3],$$

$\mathrm{Trace}(\mathrm{Blk}_i(L_2)) = 1, i = 1, \cdots, 8$, $\mathrm{Trace}(\mathrm{Blk}_9(L_2)) = 2$. 对于 $u(t+1)u(t) = \delta_9^1 = \delta_3^1\delta_3^1$,

$$T(\mathrm{Blk}_1(L_2)) = \mathrm{Trace}(\mathrm{Blk}_1(L_2)) - \mathrm{Trace}(\mathrm{Blk}_1(L_1)) = 0.$$

类似地, 有 $T(\mathrm{Blk}_1(L_2)) = T(\mathrm{Blk}_5(L_2)) = T(\mathrm{Blk}_9(L_2)) = 0$, $T(\mathrm{Blk}_i(L_2)) = 1, i = 2, 3, 4, 6, 7, 8$. 因此, $N_2 = 3$. δ_3^2 处于在控制 $u(t+1)u(t) = \delta_9^i, i = 2, 3, 4, 6, 7, 8$ 下长度为 2 的环上. 于是可以找出长度为 2 的环如下:

$$\delta_3 \times \delta_3((1,2),(2,2)),\ \delta_3 \times \delta_3((1,2),(3,2)),\ \delta_3 \times \delta_3((2,2),(3,2)),$$

$$L_3 = L_x(I_3 \otimes L_x)(I_9 \otimes L_x) = \delta_{81}[\underbrace{2\ \cdots\ 2}_{80}\ 3].$$

利用 (21.2.9) 可得 $T(\mathrm{Blk}_i(L_3)) = 1, i = 2, \cdots, 13, 15, \cdots, 26$, $T(\mathrm{Blk}_i(L_3)) = 0, i = 1, 14, 27$, 且 $N_3 = 8$. δ_3^2 处于在控制 $u(t+2)u(t+1)u(t) = \delta_{27}^i, i = 2, \cdots, 13, 15, \cdots, 26$ 下长度为 3 的环上. 于是可以找出长度为 3 的环如下:

$$\begin{aligned}
&\delta_3 \times \delta_3((1,2),(1,2),(2,2)), \quad && \delta_3 \times \delta_3((1,2),(3,2),(2,2)),\\
&\delta_3 \times \delta_3((1,2),(1,2),(3,2)), \quad && \delta_3 \times \delta_3((1,2),(3,2),(3,2)),\\
&\delta_3 \times \delta_3((1,2),(2,2),(2,2)), \quad && \delta_3 \times \delta_3((2,2),(2,2),(3,2)),\\
&\delta_3 \times \delta_3((1,2),(2,2),(3,2)), \quad && \delta_3 \times \delta_3((2,2),(3,2),(3,2)).
\end{aligned}$$

系统有许多长度大于或等于 4 的环. 但我们知道, 为寻找这个博弈的最优解, 找出所有长度不超过 3 的环就够了.

考虑一报还一报策略, 从初态 $x_0 = \delta_3^3$ 出发, 其可达集为

$$R(x_0) = \{\delta_3^1\delta_3^2, \delta_3^2\delta_3^2, \delta_3^3\delta_3^2, \delta_3^1\delta_3^3, \delta_3^2\delta_3^3, \delta_3^1\delta_3^3\}.$$

利用前面的结果, 在可达集 $R(x_0)$ 内所有的环为 $\delta_3 \times \delta_3((1,2))$, $\delta_3 \times \delta_3((2,2))$, $\delta_3 \times \delta_3((3,2))$ 和 $\delta_3 \times \delta_3((3,3))$, 并且其中最优环为 $\delta_3 \times \delta_3((3,3))$. 选 $u^*(0) = \delta_3^3$, 那么,

$$G^* = \delta_3[* \ * \ * \ * \ * \ * \ * \ * \ 3],$$

这里前 8 个元素任选.

例如, 可以选

$$G^* = \delta_3[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 3],$$

它就是一报还一报策略. 可得到如下结论: 如果玩家 2 采用一报还一报策略, 则一报还一报策略也是玩家 1 的最佳策略. 由于该博弈的对称性, 显然, 如果玩家 1 采用一报还一报策略, 则一报还一报策略也是玩家 2 的最佳策略. 因此, 双方均采用的一报还一报策略是一个纳什均衡点. 也就是说, 它可达到 (R,R) 的重复均衡点, 它优于一次博弈的纳什均衡点 (M,M).

21.4 高阶逻辑控制网络的最优控制

一般地说, μ 阶逻辑动态网络可以描述成

$$\begin{cases} x_1(t+1) = f_1(x_1(t),\cdots,x_n(t)\cdots,x_1(t-\mu+1),\cdots,x_n(t-\mu+1),\\ \qquad\qquad u_1(t),\cdots,u_m(t),\cdots,u_1(t-\mu+1),\cdots,u_m(t-\mu+1)),\\ x_2(t+1) = f_2(x_1(t),\cdots,x_n(t),\cdots,x_1(t-\mu+1),\cdots,x_n(t-\mu+1),\\ \qquad\qquad u_1(t),\cdots,u_m(t),\cdots,u_1(t-\mu+1),\cdots,u_m(t-\mu+1)),\\ \qquad\qquad \vdots\\ x_n(t+1) = f_n(x_1(t),\cdots,x_n(t),\cdots,x_1(t-\mu+1),\cdots,x_n(t-\mu+1),\\ \qquad\qquad u_1(t),\cdots,u_m(t),\cdots,u_1(t-\mu+1),\cdots,u_m(t-\mu+1)),\\ y_j(t) = h_j(x_1(t),\cdots,x_n(t)), \quad j=1,\cdots,p. \end{cases} \tag{21.4.1}$$

为处理 μ 阶逻辑动态网络, 我们首先考虑如何将其转化为一阶形式. 先回忆 k 维降阶矩阵, 它可表示为

$$M_{r,k} = \delta_{k^2}[1\ \ k+2\ \ 2k+3\ \ \cdots\ \ (k-1)k+k]. \tag{21.4.2}$$

设 $x = \ltimes_{i=1}^{\mu} x_i \in \Delta_{k^\mu}$, 那么不难证明

$$x^2 = \Phi_{\mu,k} x, \tag{21.4.3}$$

这里

$$\Phi_{\mu,k} := \prod_{i=1}^{\mu} I_{k^{i-1}} \otimes \left[\left(I_k \otimes W_{[k,k^{\mu-i}]}\right) M_{r,k}\right] = M_{r,k^\mu}. \tag{21.4.4}$$

根据命题 14.1.1, 有如下的因子析取公式:

引理 21.4.1 设 $x = \ltimes_{i=1}^{n} x_i \in \Delta_{k^n}$, 这里 $x_i \in \Delta_k$, 那么

$$D_r[k^m, k^{n-m}]x = \ltimes_{i=1}^{m} x_i, \quad D_f[k^m, k^{n-m}]x = \ltimes_{i=n-m+1}^{n} x_i. \tag{21.4.5}$$

证明 如果 $\ltimes_{i=1}^{m} x_i = \delta_{k^m}^{j}$, 那么,

$$F_{[m,n],k} x = (I_{k^m} \otimes \mathbf{1}_{k^{n-m}}^{T}) \ltimes \delta_{k^m}^{j} \ltimes_{i=m+1}^{n} x_i = \delta_{k^m}^{j} \ltimes \mathbf{1}_{k^{n-m}}^{T} \ltimes_{i=m+1}^{n} x_i = \delta_{k^m}^{j}.$$

不管 $\ltimes_{i=1}^{n-m} x_i$ 是什么, 总有

$$E_{[m,n],k} x = E_{[m,n],k} \ltimes_{i=1}^{n-m} x_i \ltimes_{i=n-m+1}^{n} x_i = I_{k^m} \ltimes_{i=n-m+1}^{n} x_i = \ltimes_{i=n-m+1}^{n} x_i.$$

□

为方便计, 后面以 M_r 和 Φ_μ 分别代替 $M_{r,k}$ 和 $\Phi_{\mu,k}$, 约定的 k 代表逻辑类型(逻辑变量取值个数).

记 $x(t) = \ltimes_{i=1}^{n} x_i(t)$, $u(t) = \ltimes_{i=1}^{m} u_i(t)$. μ 阶逻辑控制网络 (21.4.1) 可写成分量代数形式如下:

$$\begin{cases} x_1(t+1) = M_1 u(t-\mu+1)\cdots u(t) x(t-\mu+1)\cdots x(t), \\ x_2(t+1) = M_2 u(t-\mu+1)\cdots u(t) x(t-\mu+1)\cdots x(t), \\ \qquad \vdots \\ x_n(t+1) = M_n u(t-\mu+1)\cdots u(t) x(t-\mu+1)\cdots x(t). \end{cases} \tag{21.4.6}$$

将 (21.4.6) 中各方程两边分别相乘可得

$$x(t+1) = L \ltimes_{i=1}^{\mu} u(t-\mu+i) \ltimes_{i=1}^{\mu} x(t-\mu+i), \tag{21.4.7}$$

这里

$$L = M_1 \prod_{j=2}^{n} \left[(I_{k^{\mu(m+n)}} \otimes M_j) \Phi_{\mu(m+n)} \right].$$

定义 $z(t) = \ltimes_{i=t}^{t+\mu-1} x(i)$, $v(t) = \ltimes_{i=t}^{t+\mu-1} u(i)$, 那么 (21.4.7) 可写为

$$x(t+1) = L v(t-\mu+1) z(t-\mu+1).$$

于是可知

$$
\begin{aligned}
z(t+1) &= \ltimes_{i=t+1}^{t+\mu} x(i) \\
&= \ltimes_{i=t+1}^{t+\mu-1} x(i) L v(t) z(t) \\
&= (I_{k^{(\mu-1)n}} \otimes L) \ltimes_{i=t+1}^{t+\mu-1} x(i) v(t) z(t) \\
&= (I_{k^{(\mu-1)n}} \otimes L) W_{[k^{\mu m+n}, k^{(\mu-1)n}]} v(t) z(t) \ltimes_{i=t+1}^{t+\mu-1} x(i) \\
&= (I_{k^{(\mu-1)n}} \otimes L) W_{[k^{\mu m+n}, k^{(\mu-1)n}]} v(t) x(t) \Phi_{(\mu-1)n} \ltimes_{i=t+1}^{t+\mu-1} x(i) \\
&:= \widetilde{L} v(t) z(t),
\end{aligned}
\tag{21.4.8}
$$

这里

$$
\widetilde{L} = (I_{k^{(\mu-1)m}} \otimes L) W_{[k^{\mu m+n}, k^{(\mu-1)n}]} (I_{k^{\mu m+n}} \otimes \Phi_{(\mu-1)n}).
$$

注意, 这里 $v(t), t=0,1,\cdots$ 不是完全独立的, 它们应当满足

$$
D_r[k^{(\mu-1)m},\ k^{\mu m}] v(t+1) = D_f[k^{(\mu-1)m}, k^{\mu m}] v(t).
$$

因此, (21.4.7) 可表示为

$$
\begin{cases}
z(t+1) = \widetilde{L} v(t) z(t), \\
F_{[(\mu-1)m, \mu m]} v(t+1) = D_f[k^{(\mu-1)m}, k^{\mu m}] v(t).
\end{cases}
\tag{21.4.9}
$$

类似 (21.2.5), 如果 $z(t)$ 位于长度为 d 的环上, 则有

$$
z(t) = z(t+d) = \widetilde{L}_d v(t+d-1) v(t+d-2) \cdots v(t) z(t), \tag{21.4.10}
$$

这里

$$
\widetilde{L}_d = \prod_{i=1}^{d} (I_{k^{(i-1)\mu m}} \otimes \widetilde{L}).
$$

$v(t+d-1)v(t+d-2)\cdots v(t)$ 可被简化为

$$
\begin{aligned}
& v(t+d-1) \cdots v(t) \\
=& \ltimes_{i=t+d-1}^{t+d+\mu-2} u(i) \ltimes_{i=t+d-2}^{t+d+\mu-3} u(i) \cdots \ltimes_{i=t}^{t+\mu-1} u(i) \\
=& W_{[k^{\mu m}]} \ltimes_{i=t+d-2}^{t+d+\mu-3} u(i) \ltimes_{i=t+d-1}^{t+d+\mu-2} u(i) \ltimes_{i=t+d-3}^{t+d+\mu-4} u(i) \cdots \ltimes_{i=t}^{t+\mu-1} u(i) \\
=& W_{[k^{\mu m}]} \left(u(t+d-2) \Phi_{(\mu-1)m} \ltimes_{i=t+d-1}^{t+d+\mu-3} u(i) u(t+d+\mu-2) \right) \\
& \ltimes_{i=t+d-3}^{t+d+\mu-4} u(i) \cdots \ltimes_{i=t}^{t+\mu-1} u(i) \\
=& W_{[k^{\mu m}]} (I_{k^m} \otimes \Phi_{(\mu-1)m}) \ltimes_{i=t+d-2}^{t+d+\mu-2} u(i) \ltimes_{i=t+d-3}^{t+d+\mu-4} u(i) \cdots \ltimes_{i=t}^{t+\mu-1} u(i) \\
& \vdots
\end{aligned}
$$

$$
\begin{aligned}
&= \prod_{i=1}^{d-1} \left(W_{[k^{\mu m},k^{(\mu+i-1)m}]}(I_{k^m} \otimes \Phi_{(\mu-1)m})\right) \ltimes_{i=t}^{t+\mu+d-2} u(i) \\
&:= R \ltimes_{i=t}^{t+\mu+d-2} u(i),
\end{aligned}
$$

这里

$$
R = \prod_{i=1}^{d-1} \left(W_{[k^{\mu m},k^{(\mu+i-1)m}]}(I_{k^m} \otimes \Phi_{(\mu-1)m})\right).
$$

并且, $v(t+d) = v(t)$ 必须满足, 即

$$
\ltimes_{i=t+d}^{t+d+\mu-1} u(i) = \ltimes_{i=t}^{t+\mu-1} u(i). \tag{21.4.11}
$$

设 $\mu = sd+r$, 这里 $s = [\frac{\mu}{d}]$, $\mu = r \pmod d$, 那么, 乘积 $v(t+d-1)v(t+d-2)\cdots v(t)$ 变为

$$
\begin{aligned}
&v(t+d-1)\cdots v(t) \\
&= R \ltimes_{i=t}^{t+d-1} u(i) \ltimes_{i=t+d}^{t+d+\mu-2} u(i) \\
&= R \ltimes_{i=t}^{t+d-1} u(i) \left(\ltimes_{i=t}^{t+d-1} u(i)\right)^{s-1} \ltimes_{i=t}^{t+d+r-2} u(i) \\
&= \begin{cases}
R(\Phi_{dm})^{s-1}(I_{k^{(d-1)m}} \otimes W_{[k^{(d-1)m},k^m]})\Phi_{(d-1)m} \ltimes_{i=t}^{t+d-1} u(i), & r=0, \\
R(\Phi_{dm})^{s} \ltimes_{i=t}^{t+d-1} u(i), & r=1, \\
R(\Phi_{dm})^{s}(I_{k^{(r-1)m}} \otimes W_{[k^{(r-1)m},k^{(d-r+1)m}]})\Phi_{(r-1)m} \ltimes_{i=t}^{t+d-1} u(i), & 2 \leqslant r \leqslant d-1.
\end{cases}
\end{aligned}
$$

于是, (21.4.10) 可写成

$$
z(t) = \Psi_d \ltimes_{i=t}^{t+d-1} u(i) z(t), \tag{21.4.12}
$$

这里

$$
\Psi_d = \begin{cases}
\widetilde{L}_d R(\Phi_{dm})^{s-1}(I_{k^{(d-1)m}} \otimes W_{[k^{(d-1)m},k^m]})\Phi_{(d-1)m}, & r=0, \\
\widetilde{L}_d R(\Phi_{dm})^{s}, & r=1, \\
\widetilde{L}_d R(\Phi_{dm})^{s}(I_{k^{(r-1)m}} \otimes W_{[k^{(r-1)m},k^{(d-r+1)m}]})\Phi_{(r-1)m}, & 2 \leqslant r \leqslant d-1.
\end{cases}
$$

注意到在 (21.4.12) 中, $u(i), i = t, t+1, \cdots, t+d-1$ 是独立的. 回顾在上一节中介绍的方法, 可以通过应用 (21.4.12) 并检验 $\mathrm{Blk}_i(\Psi_d)$ 的迹, 来寻找 (21.4.9) 的长度为 d 的环. 如果 $\mathrm{Blk}_i(\Psi_d)$ 的 (j,j) 元等于 1, $z(t) = \delta_{k^{\mu n}}^j$ 处于某个在控制 $u(t)u(t+1)\cdots u(t+d-1) = \delta_{k^{dm}}^i$ 下长度为 d 的环上. 应用 (21.4.11), 可以找到 $v(t), \cdots, v(t+d-1)$, 进而找到那个环. 注意到如果 ℓ 是 d 的真因子, 则 $z(t)$ 同时处于一个在控制 $\ltimes_{\xi=t}^{t+\ell-1} \widetilde{u}(\xi) = \delta_{k^{\ell m}}^j$ 下长度为 ℓ 的一个环上以及在控制 $\ltimes_{\xi=t}^{t+d-1} u(\xi) =$

$\delta^i_{k^{dm}}$ 下长度为 d 的一个环上. 这时我们得到同一个输入-状态空间的环, 当且仅当 $\delta^i_{k^{dm}}=(\delta^j_{k^{\ell m}})^{\frac{d}{\ell}}$. 要计算环的数量, 必须刨去重复计算的环. 类似定理 21.2.1, 有如下定理:

定理 21.4.1 逻辑控制网络 (21.4.9) 具有的长度为 d 的环的个数可用下式递推计算:

$$N_d=\frac{1}{d}\sum_{i=1}^{k^{dm}}T(\mathrm{Blk}_i(\Psi_d)), \tag{21.4.13}$$

这里

$$T(\mathrm{Blk}_i(\Psi_d))=\mathrm{Trace}(\mathrm{Blk}_i(\Psi_d))-\sum_{(\ell,j)\in\theta(i,d)}T(\mathrm{Blk}_j(\Psi_\ell)).$$

命题 21.4.1 系统 (21.4.9) 的环与高价逻辑动态网络 (21.4.1) 的环一一对应.

证明 因为 $\delta^i_{k^{\mu(m+n)}}$ 可唯一地分解为 $\ltimes_{\ell=1}^{\mu}\delta^{i(\ell)}_{k^m}\ltimes_{\ell=1}^{\mu}\delta^{j(\ell)}_{k^n}$, 我们可以构造一个函数 $\pi:\Delta_{k^{\mu(m+n)}}\to\Delta_{k^{m+n}}$ 如下:

$$\pi(\delta^i_{k^{\mu(m+n)}}):=D_r[k^m,k^{\mu m}](I_{k^{\mu m}}\otimes D_r[k^n,k^{\mu n}]\delta^i_{k^{\mu(m+n)}}=\delta^{i(1)}_{k^m}\delta^{j(1)}_{k^n}. \tag{21.4.14}$$

分别用 Ω_{vz} 及 Ω_{ux} 记系统 (21.4.9) 及高阶逻辑控制网络 (21.4.1) 的环的集合. 定义映射 $\psi:\Omega_{vz}\to\Omega_{ux}$ 如下: 对每个 $C=(v(t)z(t),\cdots,v(t+d-1)z(t+d-1))$, 定义

$$\psi(C):=(\pi(v(t)z(t)),\cdots,\pi(v(t+d-1)z(t+d-1))). \tag{21.4.15}$$

注意到当 $\xi=\ell\ (\mathrm{mod}\ d)$ 时, 则 $u(\xi)=u(\ell)$, $x(\xi)=x(\ell)$, $v(\xi)=v(\ell)$, $z(\xi)=z(\ell)$. 由于

$$L\ltimes_{i=t+d-\mu}^{t+d-1}\pi(v(i)z(i))=Lv(t+d-\mu)z(t+d-\mu)=F_{[n,\mu n]}z(t+d),$$

$\psi(C)$ 是 Ω_{ux} 中的一个环. 因此, ψ 是定义好的. 下面证明

(i) ψ 是映上的. 对任意一个环 $C\in\Omega_{ux}$, $C=(u(t)x(t),u(t+1)x(t+1),\cdots,u(t+d-1)x(t+d-1))$, 令 $C_1=\{v(t)z(t),\cdots,v(t+d-1)z(t+d-1)\}$, 这里 $v(i)=\ltimes_{\xi=i}^{i+d-1}u(\xi)$, $z(i)=\ltimes_{\xi=i}^{i+d-1}x(\xi)$. 那么, 不难验证 $\psi(C_1)=C$.

(ii) ψ 是一对一的. 设存在另一个环 $C_2=(\widetilde{v}(t)\widetilde{z}(t),\cdots,\widetilde{v}(t+d-1)\widetilde{z}(t+d-1))$ 使得 $\psi(C_2)=C$. 那么, 就存在一个 $a\leqslant d$, 使得 $\pi(\widetilde{v}(i)\widetilde{z}(i))=u(a+i-1)x(a+i-1)$. 即 $\widetilde{v}(i)$ 开始的 m 个因子形成 $u(a+i-1)$, 同时, $\widetilde{z}(i)$ 开始的 n 个因子形成 $x(a+i-1)$. 根据 (21.4.9) 可知 $\widetilde{v}(i+1)$ 开始的 $(k-1)m$ 因子等于 $\widetilde{v}(i)$ 最后的 $(k-1)m$ 因子. 同时, $\widetilde{z}(i+1)$ 开始的 $(k-1)n$ 因子等于 $\widetilde{z}(i)$ 最后 $(k-1)n$ 个因子. 于是有

$$\widetilde{v}(i) = \ltimes_{\xi=a+i-1}^{a+i+\mu-2} u(\xi) = v(a+i-1),$$
$$\widetilde{z}(i) = \ltimes_{\xi=a+i-1}^{a+i+\mu-2} x(\xi) = z(a+i-1).$$

由此可知, $C_2 = C_1$. □

下面讨论 μ 阶逻辑网络的最优控制. 令

$$\widetilde{J}(v) = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} \widetilde{P}(z(t), v(t)), \tag{21.4.16}$$

这里

$$\widetilde{P}(z(t), v(t)) = P(D_r[k^n, k^{\mu n}]z(t), D_r[k^m, k^{\mu m}]v(t)).$$

根据引理 21.4.1 易知 $D_r[k^n, k^{\mu n}]z(t) = x(t)$, $D_r[k^m, k^{\mu m}]v(t) = u(t)$, 于是, 最大化 (21.4.16) 等价于最大化 (21.1.4).

虽然命题 21.3.2 不再成立, 但不难看出最优环 (在 Ω_{vz} 中) 可以在无重复元素的环中找到. 因此只要寻找这样的环, 其长度不大于从 z_0 出发的可达集 $R(z_0)$ 的元素个数. 于是, 不难得到以下定理:

定理 21.4.2 考察 μ 阶逻辑控制网络 (21.4.1) 及目标函数 (21.4.16). 则存在一个最优逻辑控制矩阵 G^*, 使 (21.3.7) 成立. 而且, 最优轨线 $s^*(t) = u^*(t)x^*(t)$ 在有限时间之后变为周期函数.

证明 我们可以用 (21.4.9) 和 (21.4.16) 分别代替 (21.4.1) 和 (21.1.4) 以寻找最优控制. (21.4.9) 可以描述成具有有限顶点的有向图, 因此, 与推论 21.1.1 类似, 可以找出 Ω_{vz} 中的最优环. 然后利用 (21.4.15), 则 Ω_{ux} 中的最优环也能得到. 记从初态 $\delta_{k^n}^{j(0)}, \cdots, \delta_{k^n}^{j(\mu-1)}$ 到达 C^* 的最短路径为

$$(\delta_{k^m}^{i(0)}\delta_{k^n}^{j(0)}, \delta_{k^m}^{i(1)}\delta_{k^n}^{j(1)}, \cdots, \delta_{k^m}^{i(T_0-1)}\delta_{k^n}^{j(T_0-1)}, C^*),$$

这里

$$C^* = \delta_{k^m} \times \delta_{k^n}((i(T_0), j(T_0)), (i(T_0+1), j(T_0+1)), \cdots, (i(T_0+d-1), j(T_0+d-1))).$$

下面采用这个约定: 设 $\ell = \xi \pmod d$, 则当 $\ell \geqslant T_0$ 以及 $T_0 \leqslant \xi \leqslant T_0+d-1$ 时, 令 $i(\ell) = i(\xi)$. 利用这个约定可以找到 G^*, 它满足

$$\mathrm{Col}_s(G^*) = \delta_{k^m}^{i(\ell+1)}, \quad \mu-1 \leqslant \ell \leqslant T_0+d+\mu-2, \tag{21.4.17}$$

这里

$$s = \sum_{\xi=1}^{\mu}(i(\ell-\mu+\xi)-1)k^{(\mu-\xi)m+\mu n} + \sum_{\zeta=1}^{\mu-1}(j(\ell-\mu+\zeta)-1)k^{(\mu-\zeta)n} + j(\ell),$$

而 G^* 的其他列 $(\mathrm{Col}(G^*) \subset \Delta_{k^m})$ 可以是任意的. 然后, 最优逻辑控制系统 (21.4.1) 变为

$$
\begin{cases}
x^*(t+1) = Lu^*(t-k+1)\cdots u^*(t)x^*(t-k+1)\cdots x^*(t), \\
u^*(t+1) = G^*u^*(t-k+1)\cdots u^*(t)x^*(t-k+1)\cdots x^*(t).
\end{cases}
\tag{21.4.18}
$$

□

例 21.4.1　考虑无穷囚徒困境模型[113].
玩家 1 为机器, 玩家 2 是人. 他们的策略为

$$
0:\quad \text{合作};\quad 1:\quad \text{背叛}.
$$

支付双矩阵见表 21.4.1.

表 21.4.1　囚徒困境支付双矩阵

$P_1\backslash P_2$	0	1
0	3,3	0,5
1	5,0	1,1

假定机器的策略依赖于 μ 个历史记忆, 其演化规则为

$$
\begin{aligned}
m(t+1) = f_m(&m(t-\mu+1), m(t-\mu+2), \cdots, m(t), \\
&h(t-\mu+1), h(t-\mu+2), \cdots, h(t)),
\end{aligned}
\tag{21.4.19}
$$

这里, 机器策略 $m(t)$ 被视为状态, f_m 为确定的逻辑函数. 人的策略, $h(t)$, 被视为控制.

记 $p_h(t) := p_h(m(t), h(t))$ 为人的支付函数. 我们的目标是设计最优控制以最大化平均支付

$$
J = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} p_h(t).
\tag{21.4.20}
$$

假定机器的策略是 "一报还两报", 即它取 0 ($m(t+1) = 0$) 仅在对方两次取 0 之后 (只在 $(h(t-1), h(t), m(t-1), m(t)) = (0,0,1,1)$ 之后). 设初始状态为 $m(0) = m(1) = h(0) = h(1) = 0$, 那么, (21.4.19) 及人的支付 P_h 可分别改写成

$$
m(t+1) = Lh(t-1)h(t)m(t-1)m(t),
\tag{21.4.21}
$$

这里

$$
L = \delta_2[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 1\ 1\ 1];
$$

以及

$$P(m(t),h(t)) := P_h = h^{\mathrm{T}}(t)\begin{bmatrix}1 & 5\\0 & 3\end{bmatrix}m(t).$$

令 $z(t) = m(t)m(t+1)$, $v(t) = h(t)h(t+1)$, $s(t) = v(t)z(t)$, 根据 (21.4.8), (21.4.21) 可写成

$$z(t+1) = \widetilde{L}v(t)z(t), \tag{21.4.22}$$

这里

$$\begin{aligned}\widetilde{L} &= (I_2 \otimes L_m)W_{[8,2]}(I_8 \otimes MR)\\ &= \delta_4[1\ 3\ 1\ 3\ 1\ 3\ 1\ 3\ 1\ 3\ 1\ 3\ 2\ 3\ 1\ 3].\end{aligned}$$

由 (21.4.16),

$$\begin{aligned}&\widetilde{P}(\delta_4^1\delta_4^1) = \widetilde{P}(\delta_4^1\delta_4^2) = \widetilde{P}(\delta_4^2\delta_4^1) = \widetilde{P}(\delta_4^2\delta_4^2) = 1;\\ &\widetilde{P}(\delta_4^1\delta_4^3) = \widetilde{P}(\delta_4^1\delta_4^4) = \widetilde{P}(\delta_4^2\delta_4^3) = \widetilde{P}(\delta_4^2\delta_4^4) = 5;\\ &\widetilde{P}(\delta_4^3\delta_4^1) = \widetilde{P}(\delta_4^3\delta_4^2) = \widetilde{P}(\delta_4^4\delta_4^1) = \widetilde{P}(\delta_4^4\delta_4^2) = 0;\\ &\widetilde{P}(\delta_4^3\delta_4^3) = \widetilde{P}(\delta_4^3\delta_4^4) = \widetilde{P}(\delta_4^4\delta_4^3) = \widetilde{P}(\delta_4^4\delta_4^4) = 3.\end{aligned}$$

容易算出, 由 $(x(0),u(0),x(1),u(1)) = (0,0,0,0)$ 出发的可达集为

$$R(\delta_4^4\delta_4^4) = \{\delta_4^1\delta_4^1, \delta_4^1\delta_4^3, \delta_4^2\delta_4^1, \delta_4^2\delta_4^3, \delta_4^3\delta_4^1, d_4^3\delta_4^2, \delta_4^3\delta_4^3, \delta_4^4\delta_4^1, \delta_4^4\delta_4^2, \delta_4^4\delta_4^3\},$$

它含 10 个元素.

利用定理 21.4.1, 可以找出长度不超过 10 的无重复元素环如下:

$$\begin{cases}C_1 = \delta_4 \times \delta_4\{(1,1)\},\\ C_2 = \delta_4 \times \delta_4\{(2,1),(3,1)\},\\ C_3^1 = \delta_4 \times \delta_4\{(1,1),(2,1),(3,1)\},\\ C_3^2 = \delta_4 \times \delta_4\{(2,3),(4,1),(3,2)\},\\ C_3^3 = \delta_4 \times \delta_4\{(4,1),(4,2),(4,3)\},\\ C_4^1 = \delta_4 \times \delta_4\{(1,3),(2,1),(4,1),(3,2)\},\\ C_4^2 = \delta_4 \times \delta_4\{(2,1),(4,1),(4,2),(3,3)\},\\ C_5^1 = \delta_4 \times \delta_4\{(1,1),(2,1),(4,1),(3,2),(1,3)\},\\ C_5^2 = \delta_4 \times \delta_4\{(2,1),(4,1),(3,2),(2,3),(3,1)\},\\ C_5^3 = \delta_4 \times \delta_4\{(1,1),(2,1),(4,1),(4,2),(3,3)\},\\ C_5^4 = \delta_4 \times \delta_4\{(2,1),(4,1),(4,2),(4,3),(3,1)\},\\ C_6^1 = \delta_4 \times \delta_4\{(1,1),(2,1),(4,1),(3,2),(2,3),(3,1)\},\\ C_6^2 = \delta_4 \times \delta_4\{(1,1),(2,1),(4,1),(4,2),(4,3),(3,1)\}.\end{cases}$$

直接计算可知, 最优环为 C_3^2, 它对应的平均人支付为 $\frac{5}{3}$. 这个结果与文献 [113] 一致.

系统 (21.4.22) 的最优轨道为

$$\delta_4^4\delta_4^4 \to \delta_4^4\delta_4^3 \to \delta_4 \times \delta_4\{(4,1),(3,2),(2,3)\}.$$

利用它, 可以发现 (21.4.21) 的最优轨道, 为

$$\delta_2^2\delta_2^2 \to \delta_2^2 \times \delta_2^2 \to \delta_2 \times \delta_2\{(2,1),(2,1),(1,2)\}.$$

于是

$$G^* = \delta_2[* \; * \; * \; * \; * \; * \; 2 \; * \; * \; 2 \; 1 \; * \; * \; * \; * \; 2 \; 2],$$

这里 $* \in \{1,2\}$ 可任选.

21.5 概率逻辑网络的最优控制

本章讨论概率逻辑网络的最优控制问题, 这里, 概率指逻辑函数可以以一定概率取不同的形式, 它可能来自逻辑演化的随机性[132], 博弈中的混合策略[79], 等. 显然, 本章前面讨论的方法不能用于这种情况.

21.5.1 问题的陈述与表达

定义 21.5.1 设 $x_i \in \mathcal{D}_{k_i}$, $i=1,\cdots,n$. $g: \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 称为一个概率逻辑函数, 如果存在 $g_j: \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$, $j=1,\cdots,s$, 使得 g 以 p_j 的概率取为 g_j, 即

$$P(g=g_j)=p_j, \quad j=1,\cdots,s.$$

这里 $\sum_{j=1}^s p_j = 1$.

定义 21.5.2 一个概率逻辑动态系统可表示为

$$\begin{cases} x_1(t+1) = g_1(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \\ \quad\vdots \\ x_n(t+1) = g_n(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \end{cases} \tag{21.5.1}$$

这里 $x_i \in \mathcal{D}_{k_i}$, $i=1,\cdots,n$ 为状态变量, $u_j \in \mathcal{D}_{r_j}$, $j=1,\cdots,m$ 为控制变量, $g_s: \prod_{i=1}^n \mathcal{D}_{k_i} \times \prod_{j=1}^m \mathcal{D}_{r_j} \to \mathcal{D}_{k_s}$, $s=1,\cdots,n$ 为概率逻辑函数[55].

设目标函数为

$$J=\sum_{t=0}^{T}\lambda^{t}c(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)),\tag{21.5.2}$$

这里 $0<\lambda<1$ 为折扣因子, $c:\prod_{i=1}^{n}\mathcal{D}_{k_i}\times\prod_{j=1}^{m}\mathcal{D}_{r_j}\to\mathbb{R}$ 为伪混合值逻辑函数, $0<T\leqslant\infty$.

最优化问题为: 找到最优控制 $u_i^*(t)$, $i=1,\cdots,m$, 使得

$$\begin{aligned}&J(x(0),u_1^*(t),\cdots,u_m^*(t),\ 0\leqslant t\leqslant T)\\=&\max_{u_j(t)\in\mathcal{D}_{r_j},\forall j,\forall t}J(x(0),u_1(t),\cdots,u_m(t),\ 0\leqslant t\leqslant T).\end{aligned}\tag{21.5.3}$$

假定模型中 g_i 有 s_i 不同的选择, 以给定的概率选取. 准确地说,

$$P(g_i=g_i^j)=p_i^j,\quad j=1,\cdots,s_i,\quad i=1,\cdots,n,\tag{21.5.4}$$

这里$\sum_{j=1}^{s_i}p_i^j=1$, $i=1,\cdots,n$, 则网络变为概率混合值网络, 它是概率布尔网络的自然推广[132].

控制的目的就是要极大化期望值

$$J(x(0))=E\left[\sum_{t=0}^{T}\lambda^{t}c(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t))\middle|x(0)\right].\tag{21.5.5}$$

考虑一个概率逻辑动态系统, 这里, G 满足 (21.5.4). 记 $s=\prod_{i=1}^{n}s_i$, 那么, 有 s 不同的模型 G_ℓ 依概率 p_ℓ, $\ell=1,\cdots,s$. 设 $G_\ell=\left(g_1^{\ell_1},\cdots,g_n^{\ell_n}\right)$, 那么, $p_\ell=\prod_{i=1}^{n}p_i^{\ell_i}$.

不难推出, 系统关于期望值的代数状态空间表达记作[55]

$$Ex(t+1)=L^*u(t)Ex(t),\tag{21.5.6}$$

这里 $L^*:=EL_G=\sum_{\ell=1}^{s}p_\ell L_{G_\ell}$.

注 实际上, $L^*u(t)$ 是依赖于控制的马尔可夫转移矩阵. 严格地说, 记 $L^*u:=A(u)=(a_{i,j}(u))$. 那么

$$P(x(t+1)=i|x(t)=j,u(t)=u)=a_{i,j}(u).\tag{21.5.7}$$

21.5.2　有限步最优控制

先考虑有限步 ($T = N < \infty$) 优化控制问题. 我们的目的是优化下列有限步目标函数:

$$J(x(0)) := E\left[\sum_{t=0}^{N} \lambda^t c(u(t), x(t)) \middle| x(0)\right]. \tag{21.5.8}$$

下面这个命题是基本的[24, 70].

命题 21.5.1　设 $J^*(x(0))$ 为 (21.5.8) 的最优值. 那么

$$J^*(x(0)) = J_0(x(0)), \tag{21.5.9}$$

这里, 泛函 J_0 由动态规划算法在最后一步得出. 令 $c_t := \lambda^t c(u(t), x(t))$, 那么, 算法从时间 $t = N$ 到 $t = 0$ 反步进行.

$$J_N(x(N)) = \max_{u(N)\in\Delta_r} c_t(u(N), x(N)). \tag{21.5.10}$$

并且

$$\begin{aligned} J_t(x(t)) = &\max_{u(t)\in\Delta_r} E\left[c_t(u(t), x(t)) + J_{t+1}(x(t+1))| \, x(t), u(t)\right], \\ &t = 0, 1, \cdots, N-1. \end{aligned} \tag{21.5.11}$$

注意, 根据 (21.5.7), (21.5.11) 右边的期望值是关于 $x(t)$ 和 $u(t)$ 的条件期望, 因此有

$$E\left[J_{t+1}(x(t+1))|x(t), u(t)\right] = \sum_{i=1}^{k} a_{i,x(t)}(u(t)) J_{t+1}(i). \tag{21.5.12}$$

考察指标函数 (21.5.8). 设 $c(\delta_r^i, \delta_k^j) =: \varphi_{i,j}$, $i = 1, \cdots, r$; $j = 1, \cdots, k$, 定义支付矩阵如下:

$$\Phi := (\varphi_{i,j}) \in \mathcal{M}_{r\times k}. \tag{21.5.13}$$

那么有

$$c_t = \lambda^t c(u(t), x(t)) = \lambda^t u^{\mathrm{T}}(t) \Phi x(t). \tag{21.5.14}$$

上述优化问题动态规划解变为:

$$\begin{cases} J_N(x(N)) = \displaystyle\max_{u(N)\in\Delta_r} \lambda^N u^{\mathrm{T}}(N) \Phi x(N); \\ J_t(x(t)) = \displaystyle\max_{u(t)\in\Delta_r} \left[\lambda^t u^{\mathrm{T}}(t) \Phi x(t) + \sum_{i=1}^{k} a_{i,x(t)}(u(t)) J_{t+1}(i)\right], \\ \qquad\qquad t = N-1, N-2, \cdots, 1, 0. \end{cases} \tag{21.5.15}$$

将 $J_t(x(t))$ 依不同的 $x(t) \in \{\delta_k^i | i = 1, \cdots, k\}$ 写成矩阵形式如下

$$J_t := \begin{bmatrix} J_t(\delta_k^1) \\ J_t(\delta_k^2) \\ \vdots \\ J_t(\delta_k^k) \end{bmatrix},$$

那么, (21.5.15) 的第一个方程变为

$$J_N = \lambda^N \begin{bmatrix} \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\mathrm{Col}_1(\Phi) \\ \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\mathrm{Col}_2(\Phi) \\ \vdots \\ \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\mathrm{Col}_k(\Phi) \end{bmatrix}. \tag{21.5.16}$$

注意到 $u(N) \in \Delta_r$, 设

$$\xi^i(N) := \mathrm{Col}_i(\Phi),$$

那么, 对 $J_N(x(N))\big|_{x(N)=\delta_k^i}$ 的最优控制是 $u_i^*(N) = \delta_r^{j^*}$, 这里

$$j^* = \mathrm{argmax}_j \xi_j^i(N), \tag{21.5.17}$$

这里 $\xi_j^i(N)$ 是 $\xi^i(N)$ 的第 j 个分量.

同样, 不难算出 (21.5.15) 的第二个方程如下:

$$J_t = \begin{bmatrix} \max\limits_{u(t)\in\Delta_r} \{\lambda^t u^{\mathrm{T}}(t)\mathrm{Col}_1(\Phi) + \mathrm{Row}_1[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\} \\ \max\limits_{u(t)\in\Delta_r} \{\lambda^t u^{\mathrm{T}}(t)\mathrm{Col}_2(\Phi) + \mathrm{Row}_2[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\} \\ \vdots \\ \max\limits_{u(t)\in\Delta_r} \{\lambda^t u^{\mathrm{T}}(t)\mathrm{Col}_k(\Phi) + \mathrm{Row}_k[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\} \end{bmatrix}. \tag{21.5.18}$$

注意到 $x^{\mathrm{T}}u^{\mathrm{T}}(L^*)^{\mathrm{T}} = u^{\mathrm{T}}x^{\mathrm{T}}W_{[r,k]}(L^*)^{\mathrm{T}}$, 定义

$$B := W_{[r,k]}[(L^*)^{\mathrm{T}}] =: \begin{bmatrix} B_1 \\ \vdots \\ B_k \end{bmatrix}, \tag{21.5.19}$$

这里 $B_i \in \mathcal{M}_{r\times k}$, $i = 1, \cdots, k$. 于是有

$$\mathrm{Row}_i[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}] = u(t)^{\mathrm{T}}B_i, \quad i = 1, \cdots, r. \tag{21.5.20}$$

于是, (21.5.18) 可改写成

$$J_t = \begin{bmatrix} \max\limits_{u(t)\in\Delta_r} u^{\mathrm{T}}(t)\left[\lambda^t \mathrm{Col}_1(\Phi) + B_1 J_{t+1}\right] \\ \max\limits_{u(t)\in\Delta_r} u^{\mathrm{T}}(t)\left[\lambda^t \mathrm{Col}_2(\Phi) + B_2 J_{t+1}\right] \\ \vdots \\ \max\limits_{u(t)\in\Delta_r} u^{\mathrm{T}}(t)\left[\lambda^t \mathrm{Col}_k(\Phi) + B_k J_{t+1}\right] \end{bmatrix}. \tag{21.5.21}$$

因为 $u(t) \in \Delta_r$, 设

$$\xi^i(t) = \lambda^t \mathrm{Col}_i(\Phi) + B_i J_{t+1} \in \mathbb{R}^r,$$

于是 $J_t(x(t))\big|_{x(t)=\delta_k^i}$ 的最优解为 $u_i^*(t) = \delta_r^{j^*}$, 这里

$$j^* = \mathrm{argmax}_j \xi_j^i(t). \tag{21.5.22}$$

(21.5.22) 表明最优解容易直接得到.

公式 (21.5.15)–(21.5.22) 给出了有限步优化问题完整的解.

21.6 无穷步博弈基于预测的反馈控制

预测控制被广泛应用于经典控制问题中[92]. 简述如下: 设控制系统的动力学方程已知. 我们考虑无穷步最优控制问题, 即, 找出控制序列 $\{u(t)|t = 0, 1, 2, \cdots\}$, 它优化性能指标

$$J = \sum_{t=0}^{\infty} c(u(t), x(t)) := J_0^{\infty}.$$

对一般情况, 定义 $J_i^j := \sum\limits_{t=i}^{j} c(u(t), x(t))$. 代替这个无穷步最优问题, 有穷步优化逼近建议固定一个滤波长度, 记作 ℓ. 首先找到有限步优化控制 $u_0^{\ell-1} = (u_0^{\ell-1}(0), u_0^{\ell-1}(1), \cdots, u_0^{\ell-1}(\ell-1))$, 它优化 $J_0^{\ell-1}$. 然后选 $u^*(0) = u_0^{\ell-1}(0)$ 作为实际控制. 接着考虑优化问题 $\max_{u(1),\cdots,u(\ell)} J_1^{\ell}$, 找出相应的优解控制 $u_1^{\ell} = \left(u_1^{\ell}(1), u_1^{\ell}(2), \cdots, u_1^{\ell}(\ell)\right)$. 同样, 采用 $u^*(1) = u_1^{\ell}(1)$ 作为第 2 个实际控制, 如此等等. 用这种办法, 无限步优化问题可转化为一系列有限步优化问题.

应用预测控制的思想, 重新考虑系统 (21.5.1) 的无穷步优化问题, 那里, $g_i's$ 满足 (21.5.4). 无穷步优化的性能指标记为

$$J(x(0)) := E\left[\sum_{t=0}^{\infty} \lambda^t c(u(t), x(t)) \,\middle|\, x(0)\right]. \tag{21.6.1}$$

应用预测控制, 优化

$$J_s^{s+\ell-1} = J(x(s)) := E\left[\sum_{t=s}^{s+\ell-1} \lambda^t c(u(t), x(t)\Big| x(s)\right], \tag{21.6.2}$$

对 $s = 0, 1, 2, \cdots$ 依次得到优化控制序列 $u^*(0), u^*(1), \cdots$.

定义

$$\begin{aligned} d &:= \min_{x\in\Delta_k} \min_{u_i\neq u_j\in\Delta_r} |c(x, u_i) - c(x, u_j)|; \\ M &:= \max_{u\in\Delta_r, x\in\Delta_k} |c(u, x)| < \infty. \end{aligned}$$

定理 21.6.1 设 $d > 0$. 那么, 由预测控制得到的控制序列 $u^*(0), u^*(1), \cdots$ 生成无穷步的最优控制, 如果滤波长度 ℓ 满足

$$\ell > \log_\lambda \frac{(1-\lambda)d}{2M}. \tag{21.6.3}$$

证明 利用 (21.6.3), 有

$$\left(\lambda^\ell + \lambda^{\ell+1} + \lambda^{\ell+2} + \cdots\right) M = \frac{\lambda^\ell}{1-\lambda} M < d/2. \tag{21.6.4}$$

假设 $\{u_0^{\ell-1}(0), u_0^{\ell-1}(1), \cdots, u_0^{\ell-1}(\ell-1)\}$ 是最大化 $J_0^{\ell-1} = J(x(0))$ 的最优控制序列, 它由 (21.6.2) 确定. 设 $\{u_0^\infty(0), u_0^\infty(1), \cdots\}$ 为 $\max_{u(0),u(1),\cdots} J_0^\infty$ 的最优控制. 考察一组给定的观测数据 $\{x(0), x(1), \cdots\}$. 我们证明, 如果 ℓ 满足 (21.6.3), 那么

$$0 \leqslant J_0^{\ell-1}\left(x(0), u_0^{\ell-1}(0), \cdots, u_0^{\ell-1}(\ell-1)\right) - J_0^{\ell-1}\left(x(0), u_0^\infty(0), \cdots, u_0^\infty(\ell-1)\right) < d. \tag{21.6.5}$$

实际上 "$\geqslant 0$" 来自 $\{u_0^{\ell-1}(i) | i = 0, \cdots, \ell-1\}$ 的最优性, 因此只有 "$< d$" 需要证明. 假定 (21.6.5) 不成立, 那么有

$$\begin{aligned} & J_0^\infty\left(x(0), u_0^{\ell-1}(0), \cdots, u_0^{\ell-1}(\ell-1), u_0^\infty(\ell), u_0^\infty(\ell+1), \cdots\right) \\ & -J_0^\infty\left(x(0), u_0^\infty(0), \cdots, u_0^\infty(\ell-1), \cdots\right) \\ > & \left[J_0^{\ell-1}\left(x(0), u_0^{\ell-1}(0), \cdots, u_0^{\ell-1}(\ell-1)\right) - \frac{d}{2}\right] \\ & - \left[J_0^{\ell-1}\left(x(0), u_0^\infty(0), \cdots, u_0^\infty(\ell-1)\right) + \frac{d}{2}\right] \\ \geqslant & \, d - 2\cdot\frac{d}{2} = 0, \end{aligned} \tag{21.6.6}$$

这导致与 $\{u_0^\infty(i) | i = 0, 1, \cdots\}$ 最优性的矛盾.

其次, 假定 $u_0^{\ell-1}(0) \neq u_0^{\infty}(0)$. 那么, 当 ℓ 满足 (21.6.3) 时, 有 (21.6.5). 根据动态规划原理, $\{u_0^{\infty}(1), u_0^{\infty}(2), \cdots\}$ 也是对 J_1^{∞} 的最优控制, 并且, $u_0^{\ell-1}(1)$, $u_0^{\ell-1}(2)$, $\cdots$, $u_0^{\ell-1}(\ell-1)$ 也是对 $J_1^{\ell-1}$ 的最优控制. 类似的讨论可知

$$0 \leqslant J_1^{\ell-1}\left(x(1), u_0^{\ell-1}(1), \cdots, u_0^{\ell-1}(\ell-1)\right) - J_1^{\ell-1}\left(x(1), u_0^{\infty}(1), \cdots, u_0^{\infty}(\ell-1)\right) < d. \tag{21.6.7}$$

将 (21.6.7) 代入 (21.6.5) 可得

$$\begin{aligned} 0 \leqslant & \left[c(x(0), u_0^{\ell-1}(0)) + J_1^{\ell-1}\left(x(1), u_0^{\ell-1}(1), \cdots, u_0^{\ell-1}(\ell-1)\right)\right] \\ & - \left[c(x(0), u_0^{\infty}(0)) + J_1^{\ell-1}\left(x(1), u_0^{\infty}(1), \cdots, u_0^{\infty}(\ell-1)\right)\right] < d. \end{aligned} \tag{21.6.8}$$

再由 (21.6.8) 及 (21.6.7) 可得

$$\left|c(x(0), u_0^{\ell-1}(0)) - c(x(0), u_0^{\infty}(0))\right| < d, \tag{21.6.9}$$

这与 d 的定义矛盾. 因此有

$$u^*(0) = u_0^{\ell-1}(0) = u_0^{\infty}(0). \tag{21.6.10}$$

类似地, 有

$$u^*(t) = u_t^{t+\ell-1}(t) = u_0^{\infty}(t), \quad t = 1, 2, \cdots. \tag{21.6.11}$$

□

综上可知, 预测控制算法可提供无穷步优化的最优控制. 同时, 在预测优化算法的每一步, 第 21.4 节发展起来的方法可用.

观察 (21.6.2), 容易看出, 最优控制是时不变状态反馈, 即

命题 21.6.1　*对无穷优化博弈, 最优化控制可以表示成*

$$u^*(t) = \Psi x(t), \tag{21.6.12}$$

这里 $\Psi \in \mathcal{L}_{r\times k}$ 是一个逻辑矩阵.

逻辑反馈矩阵 Ψ 可以用在第 21.3 节介绍的解长度为 $(\ell-1)$ 的预测控制的方法求得, 这里滤波长度 ℓ 满足 (21.6.3). 由于反馈律是时不变的, 于是, 只要利用基于预测控制的优化方法就可以找到对应不同 $x(0)$ 最优控制 $u^*(0)$, 于是得到 Ψ, 从而构造最优控制 $u^*(t)$.

最后, 我们讨论一个例子.

例 21.6.1　考察石头-剪刀-布, 设第一玩家 (P_1) 为人, 第二玩家 (P_2) 为机器. 支付双矩阵见表 21.6.1.

表 21.6.1 石头-剪刀-布支付双矩阵

$P_1\backslash P_2$	R	S	P
R	0, 0	1, −1	−1, 1
S	−1, 1	0, 0	1, −1
P	1, −1	−1, 1	0, 0

设机器可以随机地选择以下三种策略之一:

1) S_1: 如果机器在 t 时刻赢了, 它在 $t+1$ 时刻将不改变策略 t, 否则, 它在 $t+1$ 时刻取对手在 t 时刻的策略.

2) S_2: 机器按顺序 $R \to S \to P \to R$ 周期性地取策略.

3) S_3: 如果机器与人在 t 时刻策略相同, 机器在 $t+1$ 时刻还取这个策略, 否则, 它将取它和人在 t 时刻均未取到的策略.

约定 $R \sim \delta_3^1$, $S \sim \delta_3^2$, 以及 $P \sim \delta_3^3$, 并分别记 $x(t)$ 和 $u(t)$ 为机器和人在 t 时刻的策略. 那么, 机器策略 S_i, $i = 1,2,3$, 满足以下方程

$$x(t+1) = L_{S_i}u(t)x(t), \quad i = 1,2,3, \tag{21.6.13}$$

这里, 容易算出结构矩阵为

$$L_{S_1} = \delta_3[1\ 1\ 3\ 1\ 2\ 2\ 3\ 2\ 3]; \quad L_{S_2} = \delta_3[2\ 3\ 1\ 2\ 3\ 1\ 2\ 3\ 1]; \quad L_{S_3} = \delta_3[1\ 3\ 2\ 3\ 2\ 1\ 2\ 1\ 3].$$

下面假定, 机器取 S_1 的概率为 0.4, 取 S_2 和 S_3 的概率均为 0.3. 那么, 演化博弈的动态方程为 (21.5.6), 这里

$$\begin{aligned} L^* &= 0.4L_{S_1} + 0.3L_{S_2} + 0.3L_{S_3} \\ &= \begin{bmatrix} 0.7 & 0.4 & 0.3 & 0.4 & 0 & 0.6 & 0 & 0.3 & 0.3 \\ 0.3 & 0 & 0.3 & 0.3 & 0.7 & 0.4 & 0.6 & 0.4 & 0 \\ 0 & 0.6 & 0.4 & 0.3 & 0.3 & 0 & 0.4 & 0.3 & 0.7 \end{bmatrix}. \end{aligned}$$

设折扣因子为 $\lambda = 0.9$. 由支付双矩阵可得人的支付矩阵可被算出为

$$\Phi = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix}. \tag{21.6.14}$$

假定人是跟随者并具有完全信息, 即机器信息可观测. 那么

- 有限次情况: 令 $T = N = 3$. 利用 (21.5.17), 有 $u^*(3)\big|_{x(3)=\delta_3^1} = \delta_3^3$, $J_3(\delta_3^1) = (0.9)^3$; $u^*(3)\big|_{x(3)=\delta_3^2} = \delta_3^1$, $J_3(\delta_3^2) = (0.9)^3$; $u^*(3)\big|_{x(3)=\delta_3^3} = \delta_3^2$, $J_3(\delta_3^3) = (0.9)^3$.

于是

$$J_3 = (0.9)^3 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \tag{21.6.15}$$

利用 (21.5.22), 亦可算出 $u^*(2)\big|_{x(2)=\delta_3^1} = \delta_3^3$, $J_2(\delta_3^1) = 1.539$; $u^*(2)\big|_{x(2)=\delta_3^2} = \delta_3^1$, $J_2(\delta_3^2) = 1.539$; $u^*(2)\big|_{x(2)=\delta_3^3} = \delta_3^2$, $J_2(\delta_3^3) = 1.539$. 继续这个过程, 可得 $u^*(1)\big|_{x(1)=\delta_3^1} = \delta_3^3$, $J_1(\delta_3^1) = 2.439$; $u^*(1)\big|_{x(1)=\delta_3^2} = \delta_3^1$, $J_1(\delta_3^2) = 2.439$; $u^*(1)\big|_{x(1)=\delta_3^3} = \delta_3^2$, $J_1(\delta_3^3) = 2.439$, 以及 $u^*(0)\big|_{x(0)=\delta_3^1} = \delta_3^3$, $J_0(\delta_3^1) = 3.439$; $u^*(0)\big|_{x(0)=\delta_3^2} = \delta_3^1$, $J_0(\delta_3^2) = 3.439$; $u^*(0)\big|_{x(0)=\delta_3^3} = \delta_3^2$, $J_0(\delta_3^3) = 3.439$.

- 无限次情况: 容易检证

$$d = 1, \quad M = 1.$$

利用 (21.6.3), 有 $\ell > \log_{0.9} \dfrac{1-0.9}{2} = 28.4332$. 因此, 令 $\ell = 29$. 利用第 21.3 节给出的方法, 计算次数为 28 的预测控制最优解, 可得 $u^*(0)\big|_{x(0)=\delta_3^1} = \delta_3^3$, $u^*(0)\big|_{x(0)=\delta_3^2} = \delta_3^1$, $u^*(0)\big|_{x(0)=\delta_3^3} = \delta_3^2$. 因此, 有

$$u^*(t) = [u^*(0)\big|_{x(0)=\delta_3^1}, u^*(0)\big|_{x(0)=\delta_3^2}, u^*(0)\big|_{x(0)=\delta_3^3}]x(t) = \delta_3[3,1,2]x(t), \tag{21.6.16}$$

以及 $J_0^\infty\big|_{x(0)=\delta_3^1} = J_0^\infty\big|_{x(0)=\delta_3^2} = J_0^\infty\big|_{x(0)=\delta_3^3} = 10$.

21.7 注释与参考

本章从一般逻辑动态系统出发, 讨论演化博弈的最优控制问题. 首先给出确定性逻辑控制系统模型并证明最优轨道收敛于输入-状态空间的一个环. 然后, 基于逻辑控制网络的拓扑结构找到最优控制. 最后将最优控制表示为状态反馈控制. 接着考虑高阶逻辑控制网络, 也就是这样的网络, 它利用 μ 重状态与控制的历史来更新策略. 这类系统的优化问题靠将其转化为一阶系统来寻优. 理论结果被用于各种演化博弈模型中. 这部分内容主要基于文献 [154]. 最后一节考虑概率逻辑控制系统的优化问题. 它推广了文献 [70, 118] 中布尔网络的结果. 概率逻辑控制系统的优化方法基于文献 [66].

参 考 文 献

[1] 程代展, 齐洪胜. *矩阵的半张量积——理论与应用*. 北京: 科学出版社, 2007 (第 2 版, 2011)

[2] 程代展. *系统与控制中的近代数学基础*. 北京: 清华大学出版社, 2007 (第 2 版, 2014)

[3] 程代展, 齐洪胜, 赵寅. 矩阵的半张量积 —— 一个便捷的新工具. *科学通报*, 2012, 56(32): 2664-2674

[4] 程代展, 刘挺, 王元华. 博弈论中的矩阵方法, *系统科学与数学*, 2014, 34(11): 1291-1305

[5] 胡冠章. *应用近世数学*. 北京: 清华大学出版社, 1998

[6] 马进, 程代展, 梅生伟, 卢强. 电力系统稳定域边界基于半张量积的渐近算法, 第一部分: 理论基础. *电力系统自动化*, 2006, 30(10): 1-5

[7] 马进, 程代展, 梅生伟, 卢强. 电力系统稳定域边界基于半张量积的渐近算法, 第二部分: 应用. *电力系统自动化*, 2006, 30(11): 1-6

[8] 梅生伟, 刘锋, 薛安成. *电力系统暂态分析中的半张量积方法*. 北京: 清华大学出版社, 2010

[9] 谭春桥, 张强. *合作对策理论及应用*, 北京: 科学出版社, 2011

[10] 王树和. *数学聊斋*. 北京: 科学出版社, 2008.

[11] 温巧燕, 钮心忻, 杨义先. *现代密码学中的布尔函数*. 北京: 科学出版社, 2000

[12] 谢政. *对策论导论*. 北京: 科学出版社, 2010

[13] 张贤达. *矩阵分析及其应用*. 北京: 清华大学出版社/Springer, 2004

[14] Abouheaf M I, Lewis F L, et al.. Discrete-time dynamic graphical games: Model-free reinforcement learning solution. *Cont. Theo. Tech.*, 2015, 13(1): 55-69

[15] Agaian S S, Astola J, Egiazarian K. *Binary Polynomial Transforms and Nonlinear Digital Filters*. New York: Marcel Dekker, 1995

[16] Agaian S S, Panetta K A, Nercessian S C, Danahy E E. Boolean derivatives with application to edge detection for imaging systems. *IEEE Trans Syst., Man., Man, Cybern. B, Cubern.*, 2010, 40(2): 371-382

[17] Akers Jr. S B. On a theory of Boolean functions. *J. SIAM*, 1959, 7(4): 487-498

[18] Akutsu T, Miyano S, Kuhara S. Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. *Proc. Pacific Symposium on Biocomputing*, Singapore: World Scientific, 1999: 17-28

[19] Akutsu T, Miyano S, Kuhara S. Algorithms for identifying Boolean networks and related biological networks based on matrix multiplication and fingerprint function. *J. Computational Biology*, 2000, 7(3/4): 331-343

[20] Ashenhurst R L. The decomposition of switching functionis. *In Proceedings of an International Symposium on the Theory of Switching*, April 1957: 74-116

[21] Bates D, Watts D. Relative curvature measures of nonlinearity. *J. Royal Stat. Soc., Series B (Methodological)*, 1980, 42: 1-25

[22] Bazso F. Derivation of vector-valued Boolean functions. *Acta Math. Hungar*, 2000, 87(3): 197–203

[23] Benoit J P, Krishna V. Finitely repeated games. *Econometrica*, 1985, 17(4):317-320

[24] Bertsekas D P. *Dynamic Programming and Stochastic Control*. New York: Academic Press, 1976

[25] Bhakar R, Sriram V S, et al.. Probabilistic game approaches for network cost allocation. *IEEE Trans. Power Sys.*, 2010, 25(1): 51-58

[26] Bilbao J M. *Cooperative games on combinatorial structures*. Boston: Kluwer Acad. Pub., 2000

[27] Blume L. The statistical mechanics of strategic interaction. *Games Econ. Behav.*, 1993, 5: 387-424

[28] Bochmann D. *Boolean Differential Calculus*. Larl Marx Stadt, German Democratic Pepublic, 1978

[29] Boole G. *An Investigation of the Laws of Thought: On Which are founded the Mathematical Theories of Logic and Probabilities*. London: Walton and Maberly, 1854

[30] Boros E, Hammer P L. Pseudo-Boolean optimization. *Discrete Appl. Math.*, 2002, 123:155-225

[31] Brandt F, Fischer F, Holzer M. Symmetries and the complexity of pure Nash equilibrium. *J. Comput. Sys. Sci.*, 2009, 75(3): 163-177

[32] Branzei R, Dimitrov D, Tijs S. *Models in Cooperative Game Theory*. Germany: Springer, 2005

[33] Brown F M. *Boolean Reasoning: The Logic of Boolean Equations*. 2nd Ed.. New York: Dover Pub., 2003

[34] Candogan O, Menache I, Ozdaglar A, Parrilo P A. Flows and decompositions of games: Harmonic and potential games. *Mathematcs of Operations Research*, 2011, 36(3): 474-503

[35] Candogan O, Ozdaglar A, Parrilo P A. Dynamics in near-potential games. *Games and Econ. Behav.*, 2013, 82: 66-90

[36] Cao Z, Yang X. On finite symmetric games. 2015 (Preprint)

[37] Carlet C. Boolean functions for cryptography and errror correcting codes//Crama Y and Hammer Pl. Chapter of *Boolean Methods and Models*. Cambridge: Cambridge Univ. Press, 2010

[38] Chen H, Sun J. Global stability and stabilization of switched Boolean network with impulsive effects. *Applied Math. Comp.*, 2013, 224: 624-625

[39] Cheng D. Semi-tensor product of matrices and its applications to Morgan's problem. *Science in China*, Series F, 2001, 44(3): 195-212

[40] Cheng D. *Matrix and Polynomial Approach to Dynamics Control Systems*. Beijing: Science Press, 2002

[41] Cheng D, Dong Y. Semi-tensor product of matrices and its some applications to physics. *Methods and Applications of Analysis*, 2003, 10(4): 565-588

[42] Cheng D, Ma J. Calculation of stability region. *Proc. 42nd IEEE CDC'03*, Maui, 2003: 5615-5620

[43] Cheng D, Ma J, Lu Q, Mei S. Quadratic form of stable sub-manifold for power systems. *Int. J. Robust Nonli. Sys.*, 2004, 14: 773-788

[44] Cheng D, Hu X, Wang Y. Non-regular feedback linearization of nonlinear systems via a normal form algorithm. *Automatica*, 2004, 40(3): 439–447

[45] Cheng D, Ma J, Lu Q, Mei S. Quadratic form of stable sub-manifold for power systems. *Int. J. Robust and Nonlinear Control*, 2004, 14(9-10): 773–788

[46] D. Cheng. Semi-tensor product of matrices and its applications to dynamic systems, in New Directions and Applications in Control Theory, Lecture Notes in Control and Information Sciences, W. Dayawansa, A. Lindquist, Y. Zhou (Eds.), Springer, 2005, 61-79

[47] Cheng D. On logic-based intelligent systems. *Proc. ICCA 2005*, Budapest, 2005: 71-76

[48] Cheng D, Qi H. Matrix expression of logic and fuzzy control. *Proc. 44th IEEE CDC*, Seville, 2005: 3273-3278

[49] Cheng D. Some applications of semi-tensor product of matrix in algebra. *Comp. Math with Appl.*, 2006, 52: 1045-1066

[50] Cheng D. Semi-tensor product of matrices and its applications: A survey. *ICCM 2007*, 2007, 3: 641-668

[51] Cheng D, Yang G, Xi Z. Nonlinear systems possessing linear symmetry. *Int. J. Robust Nonlin. Contr.*, 2007, 17(1): 51-81

[52] Cheng D, Qi H. Controllability and observability of Boolean control networks. *Automatica*, 2009, 45(7): 1659-1667

[53] Cheng D, Qi H. A linear representation of dynamics of Boolean networks. *IEEE Trans. Aut. Contr.*, 2010, 55(10): 2251-2258

[54] Cheng D, Qi H. State-space analysis of Boolean networks. *IEEE Trans. Neural Networks*, 2010, 21(4): 584-594

[55] Cheng D, Qi H, Li Z. *Analysis and Control of Boolean Networks: A Semi-tensor Product Approach*. London: Springer, 2011

[56] Cheng D, Qi H, Li Z Q. Model Construction of Boolean Network via Observed Data. *IEEE Trans. Neural Networks*, 2011, 22(4):525-536

[57] Cheng D, Zhao Y. Identification of Boolean control networks. *Automatica*, 2011, 47(4): 702-710

[58] Cheng D. Disturbance decoupling of Boolean control networks. *IEEE Trans. Aut. Contr.*, 2011, 56(1): 2-10

[59] Cheng D, Qi H, Li Z Q, Liu J B. Stability and stabilization of Boolean networks. *Int. J. Robust and Nonlinear Control*, 2011, 21(2): 134-156

[60] Cheng D, Qi H, Zhao Y. *An Introduction to Semi-tensor Product of Matrices and Its Applications*. Singapore: World Scientific, 2012

[61] Cheng D, Xu T. Application of STP to cooperative games. *Proc. 10th IEEE ICCA*, Zhejiang, 2013: 1680-1685

[62] Cheng D, Xu X. Bi-decomposition of multi-valued logical functions and its applications. *Automatica*, 2003, 49(7): 1979-1985

[63] Cheng D, Xu T, Qi H. Evolutionarily stable strategy of networked evolutionary games. *IEEE Trans. Neur. Netwk. Learn. Sys.*, 2014, 25(7): 1335-1345

[64] Cheng D. On finite potential games. *Automatica*, 2014, 50(7): 1793-1801

[65] Cheng D, He F, Qi H, Xu T, He F. Modeling, analysis and control of networked evolutionary games. *IEEE Trans. Aut. Contr.*, In press. On line: http://ieeexplore. ieee. org/xpl/articleDetails. jsp?arnumber=7042754, DOI: 10.1109/TAC.2015.2404471

[66] Cheng D, Zhao Y, Xu T. Receding horizon based feedback optimization for mix-valued logical networks. *IEEE Trans. Aut. Contr.*, In press. On line: http://ieeexplore. ieee. org/xpl/articleDetails.jsp?arnumber=7079492, DOI: 10.1109/TAC.2015.2419874

[67] Choudhury M, Mohanram K. Bi-decomposition of large Boolean functions using blocking edge graphs. *Proc. 2010 IEEE/ACM Int. Conf. Comput.-aided Design*, San Jose, 2010:586-591

[68] Curtis H A. *A new approach to the design of switching circuits.* Princeton, N.J.: Van Nostrand, 1962

[69] Daniell P J. The modular difference of classes. *Bull. Amer. Math. Soc*, 1917, 23: 446-450

[70] Datta A, Choudhary A, Bittner M L, Dougherty E R. External control in Markovian genetic regulatory networks. *Machine Learning*, 2003, 52:169-191

[71] Davio M, Deschamps J P, Thayse A. *Discrete and switching functions.* New York: McGraw-Hill, 1978

[72] Dixon J D, Mortimer B. *Permutation groups.* London: Springer-Verlag, 1996

[73] Emmott S, (Editor-in-Chief). *Towards 2020 Science.* Cambridge: Microsoft Prsearch Ltd, 2006 (面向 *2020* 年的科学. 国家自然科学基金委员会政策局编译, 2008)

[74] Fornasini E, Valcher M E. Observability, reconstructibility and state observers of Boolean control networks. *IEEE Trans. Aut. Contr.*, 2013, 58(6): 1390-1401

[75] Fornasini E, Valcher M E. On the periodic trajectories of Boolean control networks. *Automatica*, 2013, 49(5): 1506-1509

[76] Fornasini E, Valcher M E. Optimal control of Boolean control networks. *IEEE Trans. Aut. Contr.*, 2013, 59(5): 1258-1270

[77] Gale D, Shapley L S. Collegy admissions and the stability of marriage. *American Math. Monthly*, 1962, 69: 9-15

[78] Gao B, Li L, Peng H, et al. Principle for performing attractor transits with single control in Boolean networks. *Physical review E*, 2013, 88(6): 062706

[79] Gibbons R. *A Primer in Game Theory.* Harlow: Pearson Education Lim., 1992

[80] Gopalakrishnan R, Marden J R, Wierman A. An architectural view of game theoretic control. *Perform. Evalu. Review*, 2011, 38(3): 31-38

[81] Gowers T. *The Princeton Companion to Mathematics.* New Jersey: Princetpmniv. Press, 2008

[82] Ledley R S. Logic and Boolean algebra in medical science. *Proc. of Conf. on Application of Undergraduate Math*, Atlanta, GA, 1973

[83] Hachtel G, Somenzi F. *Logic Synthesis and Verification*. Kluwer Academic Pub., 2000

[84] Hamilton A. *Logic for Mathematicians*. revised edn. Camblidge: Cambridge Univ. Press, 1988

[85] Heikkinen T. A potential game approach to distributed power control and scheduling. Computer Networks, 2006, 50: 2295-2311

[86] Hochma G, Margaliot M, Fornasini E. Symbolic dynamics of Boolean control networks. *Automatica*, 2013, 49(8): 2525-2530

[87] Isidori A. *Nonlinear Control Systems*. 3rd Ed.. Berlin: Springer, 1995

[88] Jacobson N. *Basic Algebra 1*. second ed.. San Francisco: Freeman, 1985

[89] Kauffman S A. *At Home in the Universe*. Oxford Univ. Press, 1995

[90] Keren O. Reduction of average path length in inary decision diagrams by spectral methods. *IEEE Computers*, 2008, 57(4): 520-531

[91] Kim K H. *Boolean Matrix Theory and Applications*. New York: Marcel Dekker Inc., 1982 (何善堉, 孔德涌等译. 布尔矩阵理论及其应用. 知识出版社, 1984)

[92] Kwon W H, Han S. *Receding Horizon Control - Model Predictive Control for state Models*. London: Springer-Verlag, 2005

[93] Laschov D, Margaliot M. A maximum principle for single-input Booolean control notworks. *IEEE Trans. Aut. Contr.*,2011, 56(4): 913-917

[94] Laschov D, Margaliot M. Controllability of Boolean control networks via the Perron-Frobenius theory. *Automatica*, 2012, 48(6): 1218-1223

[95] Laschov D, Margaliot M, Even G. Observbility of Boolean networks: a graph-theoretic approach. *Automatica*, 2013, 49(8): 2351-2362

[96] Laschov D, Margaliot M. Minimum-time control of Boolean networks. *SIAM J. Contr. Opt.*, 2013, 51(4): 2869-2892

[97] Ledley R S. Digital methods in symbolic logic. *Proc. of US Nat. Acad. Sci.*, 1955, 41(7): 498-511

[98] Li F, Sun J. Controllability of Boolean control networks with time delays in states. *Automatica*, 2011, 47(3): 603-607

[99] Li F, Sun J, Wu Q. Obvervability of Boolean control networks with state time delays. *IEEE Trans. Neural Networks*, 2011, 22(6): 948-954

[100] Li F, Sun J. Stability and stabilization of multivalued logical networks. *Nonlinear Analysisi - Real World Appl.*, 2011, 12(6): 3701-3712

[101] Li F, Sun J. Stability and stabilization of Boolean neworks with impulsive effects. *Sys. Contr. Lett.*, 2012, 61(1): 1-5

[102] Li F, Sun J. Controllability of higher order Boolean control networks. *Appl. Math. Comput.*, 2012, 219(1): 158-169

[103] Li H, Wang Y. On reachability and controllability of switched Boolean control networks. *Automatica*, 2012, 48(11): 2917-2922

[104] Li H T, Wang Y Z. Boolean derivative calculation with application to fault detection of combinational circuits via the semi-tensor product method. *Automatica*, 2012, 48(4): 688-693

[105] Li H, Wang Y. Output feedback stabilization control design for Boolean control networks. *Automatica*, 2013, 49(12): 3641-3645

[106] Li R, Yang M, Chu T. State feedback stabilization for Boolean control networks. *IEEE Trans. Aut. Contr.*, 2013, 58(7): 1853-1857

[107] Liang S, Fuhrman S, Somogyi R. REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. *Proc. Pacific Symposium on Biocomputing*, 1998, 3:18-29

[108] Liu Z, Wang Y. Disturbance becoupling of mix-valued logical networks via the semi-tensor product method. *Automatica*, 2012, 48(8): 1839-1844

[109] Marden J, Arslan G, Shamma J. Joint strategy fictitious play with inertia for potential games. *IEEE Trans. Aut. Contr.*, 2009, 54(2): 208-220

[110] Marden J R, Shamma J S. Revisiting log-linear learning: Asynchrony, completeness and bayoff-based implementation. *Games Econ. Behav.*, 2012, 75: 788-808

[111] Mishchenko A, Steinbach B, Perkowski M. An algorithm for bi-decomposition of logic functions. *Proc. 2001 IEEE/ACM 38th Design Automation Conference*, Las Vegas, 2001: 103-108

[112] Monderer D, Shapley L S. Potential games. *Games and Economic Behavior*, 1996, 14: 124-143

[113] Mu Y, Guo L. Optimization and identification in a non-equilibrium dynamic game. *Proc. CDC-CCC'09*, 2009: 5750-5755

[114] Nam D, Seo S, Kim S. An efficient top-down search algorithm for learning Boolean networks of gene expression. *Machine Learning*, 2006, 65: 229-245

[115] Nash J. Non-cooperative game. *The Annals of Mathematics*, 1951, 54(2): 286-295

[116] von Neumann J, Morgenstern O. *Theory of Games and Economic Behavior*. Princeton, New Jersey, Princeton University Press: 1944

[117] Nowak M A, May R M. Evolutionary games and spatial chaos. *Nature*, 1992, 359: 826-829

[118] Pal R, Datta A, Dougherty E R. Optimal infinite-horizon control for probabilistic Boolean networks. *IEEE Trans. Signal Proc.*, 2006, 54(6):2375-2387

[119] Peleg B, Rosenmuller J, Sudholter P. The canonical extensive form of a game form: symmetries//Alkan A, Aliprantis C, Yannelis N. *Current Trends in Economics, Studies in Economic Theory*, 8:367-387. Berlin: Springer, 1999

[120] Posthoff C, Steinbach B. *Logic functions and equations: binary models for computer science*. Dordrecht, The Netherlands: Springer, 2004

[121] Qiao Y, Qi H, Cheng D. Ledley solution of static logical systems and its applications, 2015 (Preprint)

[122] Råde L, Westergren B. *Mathematics Handbook — for Science and Engineering*. Sweden: Studentlitteratur, 1998

[123] Rasmusen E. *Games and Information: An Introduction to Game Theory*. 4th Ed.. Oxford: Basil Blachwell, 2007

[124] Reed I S. A class of multiple error-correctiong and the decoding scheme. *IRE Trans. Inform. Theory*, 1954, IT-4: 38-49

[125] Robert F. *Discrete Iterations, A Metric Study*. Translated by Rokne J. Berlin: Springer-Verlag, 1986

[126] Rosenthal R W. A class of games possessing pure-strategy Nash equilibria. *Int. J. Game Theory*, 1973, 2:65-67

[127] Roth J P, Karp R M. Minimization over Boolean graphs. IBM Journal, April 1962: 227-238

[128] Sasao T, Fujita M. *Representations of discrete functions*. Kluwer Academic Publishers, 1996

[129] Sasao T, Butler J T. On Bi-decompositions of logic functions. *Proc. Int. Worksop on Logic Synthesis*, Lake Tahoe, 1997, 2: 1-6

[130] Sasao T. *Switching Theory for Logic Synthesis*. Kluwer Academic Publishers, 1999

[131] Schneeweiss W G. *Boolean Functions, with Engineering Applications and Computer Programs*. Berlin: Springer-Verlag, 1989

[132] Shmulevich I, Dougherty E R, Kim S, Zhang W. Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 2002, 18(2): 261-274

[133] Simaan M, Cruz J B. On the Stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Control*, 1973, 11(5): 533-555

[134] Smith J M. *Evolution and the Theory of Games*. Cambridge: Cambridge Univ. Press, 1982

[135] Szabo G, Toke C. Evolutionary prisoner's dilemma game on a square lattice. *Phys. Rev. E*, 1998, 58:69

[136] Thayse A. *P-Functions and Boolean Matrix Factorization*. LNCS 175, Berlin: Springer-Verlag, 1984

[137] Tong Y, Lombeyda S, Hirani A N, Desbrun M. Discrete multiscale vector field decomposition. *ACM Trans. Graphics*, 2003, 22(3): 445-452

[138] Traulsen A, Nowak M A, Pacheco J M. Stochastic dynamics of invasion and fixation. *Phys. Rev., E*, 2006, 74:011909

[139] Tsai C. *Contributions to the Design and Analysis of Nonlinear Models*. Ph.D. Thesis, University of Minisota, 1983

[140] Tucker J H, Tapia M A, Bennett A W. Boolean integral calculus. *Appl. Math. and Comput.*, 1988, 26: 201-236

[141] Vichniac G Y. Boolean derivatives on cellular automata. *Physica D*, 1990, 45: 63-74

[142] Waldrop M M. *Complexity: The Emerging Science at the Edge of Order and Chaos*. New York: Touchstone, 1992

[143] Wang Y, Zhang C, Liu Z. A matrix approach to graph maximum stable set and coloring problems with application to multi-agent systems. *Automatica*, 2012, 48(7): 1227-1236

[144] Wang X, Xiao N, Wongpiromsarn T, Xie L, Frazzoli E, Rus D. Distributed consensus in noncooperative congestion games: an application to road pricing. *Proc. 10th IEEE Int. Conf. Contr. Aut.*, Hangzhou, China, 2013: 1668-1673

[145] Wonham W M. *Linear Multivariable Control, A Geometric Approach.* Berlin: Springer-Verlag, 1974

[146] Xu X, Hong Y. Matrix expression to model matching of asynchronous sequential machines. *IEEE Trans. Aut. Contr.*, 2013, 58(11): 2974-2979

[147] Yanushkevic S. *Logic Differential Calculus in Multi-Valued Logic Design.* Szczecin: Tech. Univ. Szczecin, 1998

[148] Yang M, Li R, Chu T G. Controller design for disturbance decoupling of Boolean control networks. *Automatica*, 2013, 49(1): 273-277

[149] Yazicioglu A Y, Egerstedt M, Shamma J S. A game theoretic approach to distributed coverage of graphs by heterogeneous mobile agents. *Est. Contr. Netw. Sys.*, 2013, 4: 309-315

[150] Young H P. The evolution of conventions. *Econometrica*, 1993, 61: 57-84

[151] Zhang G Q. Automata, Boolean matrices, and ultimate periodicity. *Inform. Comput.*, 1999, 152: 138-154

[152] Zhao Y, Gao X, Cheng D. Some applications of the matrix expression of Boolean function via semi-tensor product. *J. Graduate School of CAS*, 2012, 29(6): 743-749

[153] Zhao Y, Qi H, Cheng D. Input-state incidence matrix of Boolean control networks and its applications. *Sys. Contr. Lett.*, 2010, 59(12): 767-774

[154] Zhao Y, Li Z, Cheng D. Optimal control of logical control networks. *IEEE Trans. Aut. Contr.*, 2011, 56(8): 1766-1776

[155] Zhao D, Peng H, Li L, et al.. Novel way to research nonlinear feedback shift register. *Science China F. Inform. Sci.*, 2014, 57(9): 1-14

[156] Zhu M, Martinez S. Distributed coverage games for energy-aware mobile sensor networks. *SIAM J. Cont. Opt.*, 2013, 51(1): 1-27

索　引

J

K

L

M

N

P

Q

《现代数学基础丛书》已出版书目

(按出版时间排序)

1 数理逻辑基础(上册) 1981.1 胡世华 陆钟万 著
2 紧黎曼曲面引论 1981.3 伍鸿熙 吕以辇 陈志华 著
3 组合论(上册) 1981.10 柯 召 魏万迪 著
4 数理统计引论 1981.11 陈希孺 著
5 多元统计分析引论 1982.6 张尧庭 方开泰 著
6 概率论基础 1982.8 严士健 王隽骧 刘秀芳 著
7 数理逻辑基础(下册) 1982. 8 胡世华 陆钟万 著
8 有限群构造(上册) 1982.11 张远达 著
9 有限群构造(下册) 1982.12 张远达 著
10 环与代数 1983.3 刘绍学 著
11 测度论基础 1983.9 朱成熹 著
12 分析概率论 1984.4 胡迪鹤 著
13 巴拿赫空间引论 1984.8 定光桂 著
14 微分方程定性理论 1985.5 张芷芬 丁同仁 黄文灶 董镇喜 著
15 傅里叶积分算子理论及其应用 1985.9 仇庆久等 编
16 辛几何引论 1986.3 J.柯歇尔 邹异明 著
17 概率论基础和随机过程 1986.6 王寿仁 著
18 算子代数 1986.6 李炳仁 著
19 线性偏微分算子引论(上册) 1986.8 齐民友 著
20 实用微分几何引论 1986.11 苏步青等 著
21 微分动力系统原理 1987.2 张筑生 著
22 线性代数群表示导论(上册) 1987.2 曹锡华等 著
23 模型论基础 1987.8 王世强 著
24 递归论 1987.11 莫绍揆 著
25 有限群导引(上册) 1987.12 徐明曜 著
26 组合论(下册) 1987.12 柯 召 魏万迪 著
27 拟共形映射及其在黎曼曲面论中的应用 1988.1 李 忠 著
28 代数体函数与常微分方程 1988.2 何育赞 著
29 同调代数 1988.2 周伯壎 著

30　近代调和分析方法及其应用　1988.6　韩永生　著
31　带有时滞的动力系统的稳定性　1989.10　秦元勋等　编著
32　代数拓扑与示性类　1989.11　马德森著　吴英青　段海鲍译
33　非线性发展方程　1989.12　李大潜　陈韵梅　著
34　反应扩散方程引论　1990.2　叶其孝等　著
35　仿微分算子引论　1990.2　陈恕行等　编
36　公理集合论导引　1991.1　张锦文　著
37　解析数论基础　1991.2　潘承洞等　著
38　拓扑群引论　1991.3　黎景辉　冯绪宁　著
39　二阶椭圆型方程与椭圆型方程组　1991.4　陈亚浙　吴兰成　著
40　黎曼曲面　1991.4　吕以辇　张学莲　著
41　线性偏微分算子引论(下册)　1992.1　齐民友　许超江　编著
42　复变函数逼近论　1992.3　沈燮昌　著
43　Banach 代数　1992.11　李炳仁　著
44　随机点过程及其应用　1992.12　邓永录等　著
45　丢番图逼近引论　1993.4　朱尧辰等　著
46　线性微分方程的非线性扰动　1994.2　徐登洲　马如云　著
47　广义哈密顿系统理论及其应用　1994.12　李继彬　赵晓华　刘正荣　著
48　线性整数规划的数学基础　1995.2　马仲蕃　著
49　单复变函数论中的几个论题　1995.8　庄圻泰　著
50　复解析动力系统　1995.10　吕以辇　著
51　组合矩阵论　1996.3　柳柏濂　著
52　Banach 空间中的非线性逼近理论　1997.5　徐士英　李　冲　杨文善　著
53　有限典型群子空间轨道生成的格　1997.6　万哲先　霍元极　著
54　实分析导论　1998.2　丁传松等　著
55　对称性分岔理论基础　1998.3　唐　云　著
56　Gel'fond-Baker 方法在丢番图方程中的应用　1998.10　乐茂华　著
57　半群的 S-系理论　1999.2　刘仲奎　著
58　有限群导引(下册)　1999.5　徐明曜等　著
59　随机模型的密度演化方法　1999.6　史定华　著
60　非线性偏微分复方程　1999.6　闻国椿　著
61　复合算子理论　1999.8　徐宪民　著
62　离散鞅及其应用　1999.9　史及民　编著
63　调和分析及其在偏微分方程中的应用　1999.10　苗长兴　著

64　惯性流形与近似惯性流形　2000.1　戴正德　郭柏灵　著
65　数学规划导论　2000.6　徐增堃　著
66　拓扑空间中的反例　2000.6　汪　林　杨富春　编著
67　拓扑空间论　2000.7　高国士　著
68　非经典数理逻辑与近似推理　2000.9　王国俊　著
69　序半群引论　2001.1　谢祥云　著
70　动力系统的定性与分支理论　2001.2　罗定军　张　祥　董梅芳　编著
71　随机分析学基础(第二版)　2001.3　黄志远　著
72　非线性动力系统分析引论　2001.9　盛昭瀚　马军海　著
73　高斯过程的样本轨道性质　2001.11　林正炎　陆传荣　张立新　著
74　数组合地图论　2001.11　刘彦佩　著
75　光滑映射的奇点理论　2002.1　李养成　著
76　动力系统的周期解与分支理论　2002.4　韩茂安　著
77　神经动力学模型方法和应用　2002.4　阮炯　顾凡及　蔡志杰　编著
78　同调论——代数拓扑之一　2002.7　沈信耀　著
79　金兹堡-朗道方程　2002.8　郭柏灵等　著
80　排队论基础　2002.10　孙荣恒　李建平　著
81　算子代数上线性映射引论　2002.12　侯晋川　崔建莲　著
82　微分方法中的变分方法　2003.2　陆文端　著
83　周期小波及其应用　2003.3　彭思龙　李登峰　谌秋辉　著
84　集值分析　2003.8　李　雷　吴从炘　著
85　数理逻辑引论与归结原理　2003.8　王国俊　著
86　强偏差定理与分析方法　2003.8　刘　文　著
87　椭圆与抛物型方程引论　2003.9　伍卓群　尹景学　王春朋　著
88　有限典型群子空间轨道生成的格(第二版)　2003.10　万哲先　霍元极　著
89　调和分析及其在偏微分方程中的应用(第二版)　2004.3　苗长兴　著
90　稳定性和单纯性理论　2004.6　史念东　著
91　发展方程数值计算方法　2004.6　黄明游　编著
92　传染病动力学的数学建模与研究　2004.8　马知恩　周义仓　王稳地　靳　祯　著
93　模李超代数　2004.9　张永正　刘文德　著
94　巴拿赫空间中算子广义逆理论及其应用　2005.1　王玉文　著
95　巴拿赫空间结构和算子理想　2005.3　钟怀杰　著
96　脉冲微分系统引论　2005.3　傅希林　闫宝强　刘衍胜　著
97　代数学中的 Frobenius 结构　2005.7　汪明义　著

98　生存数据统计分析　2005.12　王启华　著
99　数理逻辑引论与归结原理(第二版)　2006.3　王国俊　著
100　数据包络分析　2006.3　魏权龄　著
101　代数群引论　2006.9　黎景辉　陈志杰　赵春来　著
102　矩阵结合方案　2006.9　王仰贤　霍元极　麻常利　著
103　椭圆曲线公钥密码导引　2006.10　祝跃飞　张亚娟　著
104　椭圆与超椭圆曲线公钥密码的理论与实现　2006.12　王学理　裴定一　著
105　散乱数据拟合的模型方法和理论　2007.1　吴宗敏　著
106　非线性演化方程的稳定性与分歧　2007.4　马　天　汪守宏　著
107　正规族理论及其应用　2007.4　顾永兴　庞学诚　方明亮　著
108　组合网络理论　2007.5　徐俊明　著
109　矩阵的半张量积:理论与应用　2007.5　程代展　齐洪胜　著
110　鞅与 Banach 空间几何学　2007.5　刘培德　著
111　非线性常微分方程边值问题　2007.6　葛渭高　著
112　戴维-斯特瓦尔松方程　2007.5　戴正德　蒋慕蓉　李栋龙　著
113　广义哈密顿系统理论及其应用　2007.7　李继彬　赵晓华　刘正荣　著
114　Adams 谱序列和球面稳定同伦群　2007.7　林金坤　著
115　矩阵理论及其应用　2007.8　陈公宁　著
116　集值随机过程引论　2007.8　张文修　李寿梅　汪振鹏　高　勇　著
117　偏微分方程的调和分析方法　2008.1　苗长兴　张　波　著
118　拓扑动力系统概论　2008.1　叶向东　黄　文　邵　松　著
119　线性微分方程的非线性扰动(第二版)　2008.3　徐登洲　马如云　著
120　数组合地图论(第二版)　2008.3　刘彦佩　著
121　半群的 S-系理论(第二版)　2008.3　刘仲奎　乔虎生　著
122　巴拿赫空间引论(第二版)　2008.4　定光桂　著
123　拓扑空间论(第二版)　2008.4　高国士　著
124　非经典数理逻辑与近似推理(第二版)　2008.5　王国俊　著
125　非参数蒙特卡罗检验及其应用　2008.8　朱力行　许王莉　著
126　Camassa-Holm 方程　2008.8　郭柏灵　田立新　杨灵娥　殷朝阳　著
127　环与代数(第二版)　2009.1　刘绍学　郭晋云　朱　彬　韩　阳　著
128　泛函微分方程的相空间理论及应用　2009.4　王　克　范　猛　著
129　概率论基础(第二版)　2009.8　严士健　王隽骧　刘秀芳　著
130　自相似集的结构　2010.1　周作领　瞿成勤　朱智伟　著
131　现代统计研究基础　2010.3　王启华　史宁中　耿　直　主编

132 图的可嵌入性理论(第二版) 2010.3 刘彦佩 著
133 非线性波动方程的现代方法(第二版) 2010.4 苗长兴 著
134 算子代数与非交换 L_p 空间引论 2010.5 许全华 吐尔德别克 陈泽乾 著
135 非线性椭圆型方程 2010.7 王明新 著
136 流形拓扑学 2010.8 马 天 著
137 局部域上的调和分析与分形分析及其应用 2011.6 苏维宜 著
138 Zakharov 方程及其孤立波解 2011.6 郭柏灵 甘在会 张景军 著
139 反应扩散方程引论(第二版) 2011.9 叶其孝 李正元 王明新 吴雅萍 著
140 代数模型论引论 2011.10 史念东 著
141 拓扑动力系统——从拓扑方法到遍历理论方法 2011.12 周作领 尹建东 许绍元 著
142 Littlewood-Paley 理论及其在流体动力学方程中的应用 2012.3 苗长兴 吴家宏 章志飞 著
143 有约束条件的统计推断及其应用 2012.3 王金德 著
144 混沌、Mel'nikov 方法及新发展 2012.6 李继彬 陈凤娟 著
145 现代统计模型 2012.6 薛留根 著
146 金融数学引论 2012.7 严加安 著
147 零过多数据的统计分析及其应用 2013.1 解锋昌 韦博成 林金官 编著
148 分形分析引论 2013.6 胡家信 著
149 索伯列夫空间导论 2013.8 陈国旺 编著
150 广义估计方程估计方法 2013.8 周 勇 著
151 统计质量控制图理论与方法 2013.8 王兆军 邹长亮 李忠华 著
152 有限群初步 2014.1 徐明曜 著
153 拓扑群引论(第二版) 2014.3 黎景辉 冯绪宁 著
154 现代非参数统计 2015.1 薛留根 著
155 三角范畴与导出范畴 2015.5 章 璞 著
156 线性算子的谱分析(第二版) 2015.6 孙 炯 王 忠 王万义 编著
157 双周期弹性断裂理论 2015.6 李 星 路见可 著
158 电磁流体动力学方程与奇异摄动理论 2015.8 王 术 冯跃红 著
159 算法数论(第二版) 2015.9 裴定一 祝跃飞 编著
160 偏微分方程现代理论引论 2016.1 崔尚斌 著
161 有限集上的映射与动态过程——矩阵半张量积方法 2015.11 程代展 齐洪胜 贺风华 著

[General Information]
书名=有限集上的映射与动态过程　矩阵半张量积方法
页数=385
SS号=13922260